

# AZURE APPLIED AI SERVICES

## What are Azure Applied AI Services?

Azure Applied AI Services are high-level services focused on empowering developers to quickly unlock the value of data by applying AI into their key business scenarios. Built on top of the AI APIs of Azure Cognitive Services, Azure Applied AI Services are optimized for critical tasks ranging from monitoring and diagnosing metric anomalies, mining knowledge from documents, enhancing the customer experience through transcription analysis, boosting literacy in the classroom, document understanding and more. Previously, companies would have to orchestrate multiple AI skills, add business logic, and create a UI to go from development to deployment for their scenario – all of which would consume time, expertise, and resources – these “scenario specific” services provide developers these benefits “out of the box”.

## Azure Form Recognizer

Enabling organizations in all industries to consume information hidden within documents to increase productivity, automate business processes and generate knowledge and insights. Azure Form Recognizer is a service that lets you build automated data processing software using machine learning technology. Identify and extract text, key/value pairs, selection marks, tables, and structure from your documents. The service outputs structured data that includes the relationships in the original file, bounding boxes, confidence and more. You quickly get accurate results that are tailored to your specific content without heavy manual intervention or extensive data science expertise. Use Form Recognizer to automate data entry in your applications and enrich your documents' search capabilities. Azure Form Recognizer is built using OCR, Text Analytics and Custom Text from Azure Cognitive Services. Form Recognizer is composed of custom document processing models, prebuilt models for invoices, receipts, IDs and business cards, and the layout model.

## Azure Metrics Advisor

Protecting organization's growth by enabling them to make the right decision based on intelligence from metrics of businesses, services and physical assets. Azure Metrics Advisor uses AI to perform data monitoring and anomaly detection in time series data. The service automates the process of applying models to your data, and provides a set of APIs and a web-based workspace for data ingestion, anomaly detection, and diagnostics - without needing to know machine learning. Developers can build AIOps, predictive maintenance, and

business monitoring applications on top of the service. Azure Metrics Advisor is built using Anomaly Detector from Azure Cognitive Services.

## Azure Cognitive Search

Unlock valuable information lying latent in all your content in order to perform an action or make decisions. Azure Cognitive Search is the only cloud search service with built-in AI capabilities that enrich all types of information to help you identify and explore relevant content at scale. Use cognitive skills for vision, language, and speech, or use custom machine learning models to uncover insights from all types of content. Azure Cognitive Search also offers semantic search capability, which uses advanced machine learning techniques to understand user intent and contextually rank the most relevant search results. Spend more time innovating and less time maintaining a complex cloud search solution. Azure Cognitive Search is built using Computer Vision and Text Analytics from Azure Cognitive Services.

## Azure Immersive Reader

Enhance reading comprehension and achievement with AI. Azure Immersive Reader is an inclusively designed tool that implements proven techniques to improve reading comprehension for new readers, language learners, and people with learning differences such as dyslexia. With the Immersive Reader client library, you can leverage the same technology used in Microsoft Word and Microsoft OneNote to improve your web applications. Azure Immersive Reader is built using Translation and Text to Speech from Azure Cognitive Services.

## Azure Bot Service

Enable rapid creation of customizable, sophisticated, conversational experiences with pre-built conversational components enabling business value right out of the box. Azure Bot Service Composer is an open-source visual authoring canvas for developers and multidisciplinary teams to build bots. Composer integrates language understanding services such as LUIS and QnA Maker and allows sophisticated composition of bot replies using language generation. Azure Bot Service is built using Speech/Telephony, LUIS, and QnA Maker from Azure Cognitive Services.

## Azure Video Analyzer

Enabling businesses to build automated apps powered by video intelligence without being a video or AI expert. Azure Video Analyzer is a service for building AI-based video solutions and applications. You can generate real-time business insights from video streams, processing data near the source and applying the AI of your choice. Record videos of interest

on the edge or in the cloud and combine them with other data to power your business decisions. Azure Video Analyzer is built using Spatial Analysis from Azure Cognitive Services. Azure Video Analyzer for Media is built using Face, Speech, Translation, Text analytics, Custom vision, and textual content moderation from Azure Cognitive Services.

## Why Azure Applied AI Services?

Azure Applied AI Services reduces the time developers need to modernize business processes from months to days. These services help you accelerate time to value for specific business scenarios through a combination of Azure Cognitive Services, task-specific AI, and business logic. Each Azure Applied AI service addresses a common need and generates new opportunities across organizations such as analyzing conversations for improved customer experiences, automating document processing for operational productivity, understanding the root cause of anomalies for protecting your organization's growth, and extracting insights from content ranging from documents to videos.

By building on top of the AI models from Azure Cognitive Services as well as providing task-specific AI models and built-in business logic, Azure Applied AI Services enable developers to quickly deploy common scenarios versus building from scratch.

### **Benefits :**

Modernize business process – Use task-specific AI to solve your scenario

Accelerate development – Go live with your AI solutions quickly

Run responsibly anywhere – Enterprise-grade responsible and secure services from the cloud to the edge

### **What is the difference between Applied AI Services and Cognitive Services?**

Both Applied AI Services and Cognitive Services are designed to help developers create intelligent apps. Cognitive Services provides general purpose AI services that serve as the core engine for Applied AI Services.

Applied AI Services builds on top of Cognitive Services while also adding task-specific AI and business logic to optimize for specific use cases so that developers spend less time designing solutions or setting up pipelines. If there isn't an Applied AI Service available to meet a specific use case, developers can also build their own solutions from scratch, using Cognitive Services as building blocks.

## Azure Bot service

Azure Bot Service is Microsoft's artificial intelligence ([AI](#)) [chatbot](#) offered as a service on the Azure cloud service marketplace.

Azure Bot Service offers the ability to add intelligent agents that are capable of conversation without having to commit the resources to develop one's own AI. The service can be added to websites, apps, email, GroupMe, [Facebook Messenger](#), Kik, Skype, Slack, Microsoft Teams, Telegram, SMS, Twilio, Cortana and [Skype for Business](#).

The platform gives bot developers a software development kit ([SDK](#)) and tools to add bots to sites in the form of Azure Bot Builder. Bots made with the Azure Bot Builder are automatically added to Microsoft's directory. Azure Bot services also supports bots coded in [Python](#), [Java](#), [Javascript](#) and [#C](#).

Azure Bot Service also offers more advanced features through Cognitive Services, such as:

- Scaling to support growth
- Ability to make smart recommendations
- Language translation
- Use of [machine vision](#) to recognize users from pictures and to moderate content.

# Development options

There is more than one way to build and deploy a chatbot. Let's take a look at some options.

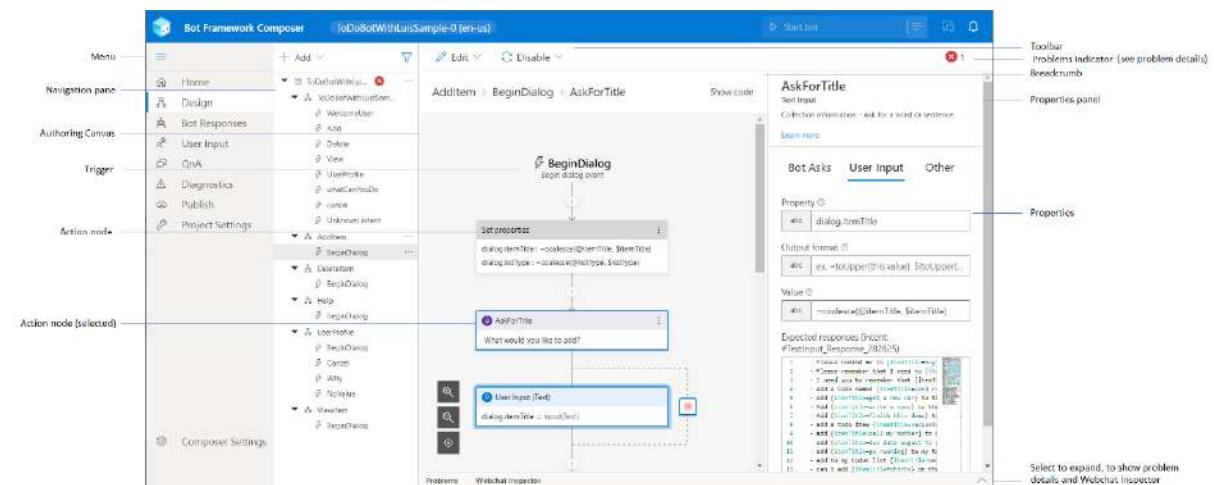
## 1. Bot Framework Composer

Bot Framework Composer, built on the Bot Framework SDK, is an open-source IDE for developers to author, test, provision, and manage conversational experiences. It provides a powerful visual authoring canvas enabling dialogs, language-understanding models, QnAMaker knowledge bases, and language generation responses to be authored from within one canvas and crucially, enables these experiences to be extended with code for more complex tasks such as system integration. Resulting experiences can then be tested within Composer and provisioned into Azure along with any dependent resources.

Composer is available as a desktop application for Windows, macOS, and Linux. If the desktop app isn't suited to your needs, you can build Composer from source or host Composer in the cloud.

Authoring dialog experiences with a visual designer is more efficient and enables easier modeling of more sophisticated conversational experiences where context switching, interruption, and more natural and dynamic conversation flows are important. More complex activities such as integrating with dependencies such as REST Web Services are best suited towards code and we provide an easy mechanism to extend Composer bots with code bringing the best of both together.

## Composer



## What you can do with Composer

Composer is a visual editing canvas for building bots. With Composer, you can:

- [Create a new bot using a template](#), which incorporates the Virtual Assistant capabilities directly into Composer.
- Add natural language understanding capabilities to your Bot using [LUIS](#) and QnA and FAQ capabilities using [QnA Maker](#).
- Author text and if needed speech variation responses for your Bot using [language generation](#) templates.
- Author bots in [multiple languages](#).
- [Test](#) directly inside Composer using embedded Web Chat.
- [Publish bots](#) to *Azure App Service* and *Azure Functions*.
- [Extend Power Virtual Agents with Composer \(Preview\)](#).
- Integrate external services such as [QnA Maker knowledge base](#).

Beyond a visual editing canvas, you can use Composer to do the following:

- Import and export dialog assets to share with other developers.
- [Package manager](#) provides a range of reusable conversational assets and code built by Microsoft and third parties. These assets can quickly add functionality to your project.
- [Make any Bot available as a Skill for other Bots to call](#).
- [Connect to a skill](#).
- Extend the dialog authoring canvas with [Create custom actions](#).
- Integrate [Orchestrator](#), which is an advanced transformer model-based router that can delegate from a parent bot to [skills](#) based on a user's utterance.
- [Host Composer in the cloud](#).
- [Extend Composer with plugins](#).

Under the hood, Composer harnesses the power of many of the components from the Bot Framework SDK. When building bots in Composer, developers will have access to:

### Adaptive dialogs

Dialogs provide a way for bots to manage conversations with users. [Adaptive dialogs](#) and the event model simplify sophisticated conversation modeling enabling more natural, dynamic conversation flow, interruption, and context switching. They also help you focus on the model of the conversation rather than the mechanics of dialog management.

## Language understanding

Language understanding is a core component of Composer that allows developers and conversation designers to train language understanding models directly in the context of editing a dialog. As dialogs are edited in Composer, developers can continuously add to their bots' natural language capabilities using the [.lu file format](#), a simple Markdown-like format that makes it easy to define new [intents](#) and [entities](#), and provide sample [utterances](#). In Composer, you can use regular expression, [LUIS](#), and [Orchestrator](#) recognizers. Composer detects changes and updates the bot's cloud-based natural language understanding model automatically so it's always up to date. Read more in the [language understanding concept article](#).

The screenshot shows two side-by-side panels in the Microsoft Bot Framework Composer.

**Create a trigger:**

- What is the type of this trigger? Intent recognized
- What is the name of this trigger (LUIS)? BuySurface
- Trigger phrases:
  - How can I buy {ProductType=Surface PRO}
  - I want to buy {ProductType=Surface PRO}
  - I want to buy {ProductType=Surface laptop}
  - Can I buy {ProductType=Surface PRO} online

**Adaptive dialog:**

- Show code
- Adaptive dialog
- This configures a data driven dialog via a collection of events and actions.
- Learn more
- Language Understanding ⓘ
- Recognizer Type: Default recognizer
- Auto end dialog ⓘ: y/n true
- Default result property ⓘ: abc dialog.result
- > Dialog Interface

## Language generation

Creating grammatically correct, data-driven responses that have a consistent tone and convey a clear brand voice has always been a challenge for bot developers. Composer's integrated bot response generation allows developers to create bot replies with a great deal of flexibility, using the editor in the [Bot Responses](#) page or the [response editor](#) in the [Properties](#) pane. Read more in the [language generation](#) concept article.

Name	Responses	Been used
#SendActivity...	- \${WelcomeUser()}	✓
#SendActivity...	- \${WelcomeUser()}	✓
<a href="#">New template</a>		

With Language Generation, you can achieve previously complex tasks easily such as:

- Including dynamic elements in messages.
- Generating grammatically correct lists, pronouns, articles.
- Providing context-sensitive variation in messages.
- Creating Adaptive Cards attachments, as seen above.
- Provide speech variations for each response, including [Speech Synthesis Markup Language \(SSML\)](#) modifications, which are key for speech-based experiences such as telephony.

## QnA Maker

[QnA Maker](#) is a cloud-based service that enables you to extract question-and-answer pairs from existing FAQ-style documents and websites into a knowledge base that can be manually curated by knowledge experts. QnA Maker, once integrated into a bot, can be used to find the most appropriate answer for any given natural language input from your custom knowledge base of information.

## Bot Framework Emulator

[Emulator](#) is a desktop application that allows bot developers to test and debug bots built using Composer. This tool allows for more advanced scenarios (like Authentication), which Composer's integrated Web Chat feature doesn't support at this time.

### Advantage of developing bots with Composer

Some of the advantages of developing bots in Composer include:

- Authoring dialogs using the visual canvas can be more conducive to a conversational design versus code and enables you to focus development efforts on more complex tasks such as system integration.
- Design conversational experiences using a seamless blend of visual and code authoring.

- Existing dialogs authored in code can be leveraged by a Composer-based bot.
- Language generation provides the ability to create more natural, personalized responses resulting in engaging conversational experiences.
- Composer streamlines your bot project's codebase and provides a more accessible visual design surface that provides a unified canvas to author dialogs and responses, along with language and QnA resources.
- Integrated testing within the Composer authoring experience.
- Azure provisioning for dependent resources is streamlined as part of the overall Composer experience.

Apps created with Composer use the declarative dialog format, a JSON specification shared by many tools provided by the Bot Framework.

The Composer bot projects contain reusable assets in the form of JSON and Markdown files that can be bundled and packaged with a bot's source code. These can be checked into source control systems and deployed along with code updates, such as dialogs, language understanding training data, and message templates.

## 2. What is the Bot Framework SDK?

The Bot Framework, along with the Azure Bot Service, provides tools to build, test, deploy, and manage intelligent bots, all in one place. The Bot Framework includes a modular and extensible SDK for building bots, as well as tools, templates, and related AI services. With this framework, developers can create bots that use speech, understand natural language, handle questions and answers, and more.

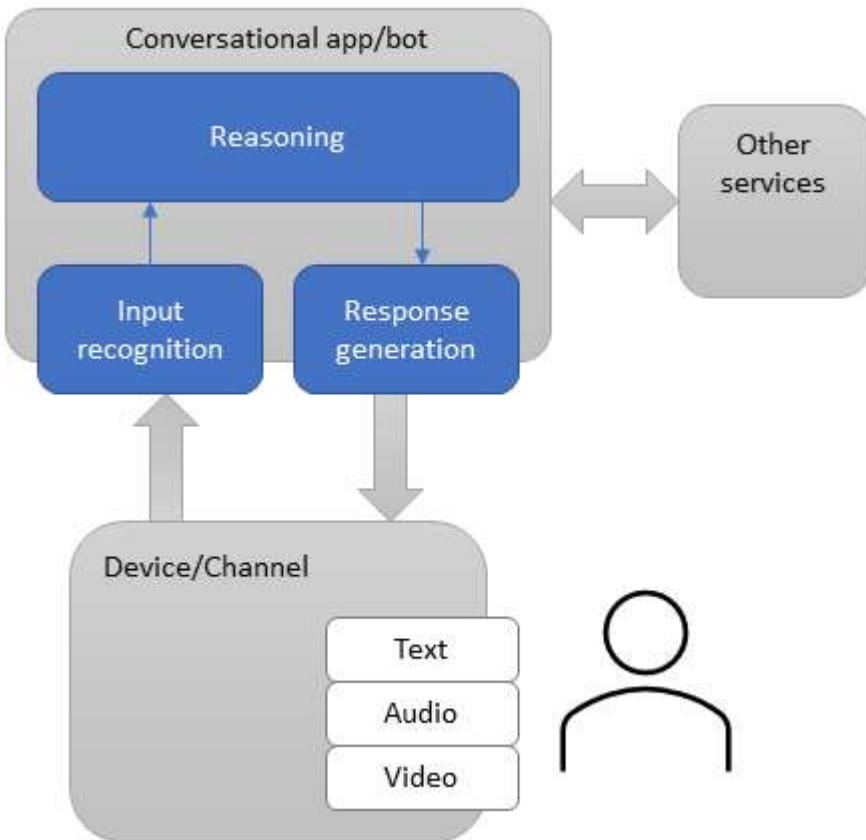
### What is a bot?

Bots provide an experience that feels less like using a computer and more like dealing with a person - or at least an intelligent robot. They can be used to shift simple, repetitive tasks, such as taking a dinner reservation or gathering profile information, on to automated systems that may no longer require direct human intervention. Users converse with a bot using text, interactive cards, and speech. A bot interaction can be a quick question and answer, or it can be a sophisticated conversation that intelligently provides access to services.

A bot can be thought of as a web application that has a conversational interface. A user connects to a bot through a channel such as Facebook, Slack, or Microsoft Teams.

- The bot *reasons* about input and performs relevant tasks. This can include asking the user for additional information or accessing services on behalf of the user.
- The bot performs recognition on the user's input to interpret what the user is asking for or saying.
- The bot generates responses to send to the user to communicate what the bot is doing or has done.

- Depending on how the bot is configured and how it is registered with the channel, users can interact with the bot through text or speech, and the conversation might include images and video.



Bots are a lot like modern web applications, living on the internet and using APIs to send and receive messages. What's in a bot varies widely depending on what kind of bot it is. Modern bot software relies on a stack of technology and tools to deliver increasingly complex experiences on a wide variety of platforms. However, a simple bot could just receive a message and echo it back to the user with very little code involved.

Bots can do the same things other types of software can do - read and write files, use databases and APIs, and do the regular computational tasks. What makes bots unique is their use of mechanisms generally reserved for human-to-human communication.

The Azure Bot Service and the Bot Framework offer:

- The Bot Framework SDK for developing bots
- Bot Framework Tools to cover end-to-end bot development workflow
- Bot Framework Service (BFS) to send and receive messages and events between bots and channels
- Bot deployment and channel configuration in Azure

Additionally, bots may use other Azure services, such as:

- Azure Cognitive Services to build intelligent applications
- Azure Storage for cloud storage solution

## How to build a bot

Azure Bot Service and the Bot Framework offer an integrated set of tools and services to facilitate this process. Choose your favorite development environment or command line tools to create your bot. SDKs exist for C#, Java, JavaScript, Typescript, and Python. We provide tools for various stages of bot development to help you design and build bots.



### Plan

As with any type of software, having a thorough understanding of the goals, processes and user needs is important to the process of creating a successful bot. Before writing code, review the bot [design guidelines](#) for best practices and identify the needs for your bot. You can create a simple bot or include more sophisticated capabilities such as speech, natural language understanding, and question answering.

### Build

Your bot is a web service that implements a conversational interface and communicates with the Bot Framework Service to send and receive messages and events. Bot Framework Service is one of the components of the Azure Bot Service and Bot Framework. You can create bots in any number of environments and languages. You can [Create a bot](#) for local development.

As part of the Azure Bot Service and Bot Framework, we offer additional components you can use to extend your bot's functionality:

BUILD			
Feature	Description	Link	
Add natural language processing	Enable your bot to understand natural language, understand spelling errors, use speech, and recognize the user's intent	How to use <a href="#">LUIS</a>	
Answer questions	Add a knowledge base to answer questions users ask in a more natural, conversational way	How to use <a href="#">QnA</a>	
Manage multiple models	If using more than one model, such as for LUIS and QnA Maker, intelligently determine when to use which one during your bot's conversation	<a href="#">Orchestrator</a>	
Add cards and buttons	Enhance the user experience with media other than text, such as graphics, menus, and cards	How to add <a href="#">cards</a>	

## Test

Bots are complex apps with a lot of different parts working together. Like any other complex app, this can lead to some interesting bugs or cause your bot to behave differently than expected. Before publishing, test your bot. We provide several ways to test bots before they are released for use:

- Test your bot locally with the [emulator](#). The Bot Framework Emulator is a stand-alone app that not only provides a chat interface but also debugging and interrogation tools to help understand how and why your bot does what it does. The Emulator can be run locally alongside your in-development bot application.
- Test your bot on the [web](#). Once configured through the Azure portal your bot can also be reached through a web chat interface. The web chat interface is a great way to grant access to your bot to testers and other people who do not have direct access to the bot's running code.
- [Unit Test](#) your bot with the current Bot Framework SDK.

## Publish

When you are ready for your bot to be available on the web, publish your bot to [Azure](#) or to your own web service or data center. Having an address on the public internet is the first step to your bot coming to life on your site, or inside chat channels.

## Connect

Connect your bot to channelssuch as Facebook, Messenger, Kik, Slack, Microsoft Teams, Telegram, text/SMS, and Twilio. Bot Framework does most of the work necessary to send and receive messages from all of these different platforms - your bot application receives a unified, normalized stream of messages regardless of the number and type of channels it is connected to. For information on adding channels, see [channels](#) topic.

## Evaluate

Use the data collected in Azure portal to identify opportunities to improve the capabilities and performance of your bot. You can get service-level and instrumentation data like traffic, latency, and integrations. Analytics also provides conversation-level reporting on user, message, and channel data

### 3. Power Virtual Agents overview

Power Virtual Agents lets you create powerful chatbots that can answer questions posed by your customers, other employees, or visitors to your website or service.

These bots can be created easily without the need for data scientists or developers. Some of the ways that Power Virtual Agents bots have been used include:

- COVID-19 infection rate and tracking information
- Sales help and support issues
- Opening hours and store information
- Employee health and vacation benefits
- Common employee questions for businesses

Power Virtual Agents is available as both a standalone web app, and as a discrete app within Microsoft Teams. Most of the functionality between the two is the same. However, there might be different reasons to choose one version or the other based on the ways you want to use Power Virtual Agents.

Power Virtual Agents is a Core Online Service, as defined in the [Online Services Terms \(OST\)](#), and is compliant with or covered by:

- Health Insurance Portability and Accountability Act (HIPAA) coverage
- Health Information Trust Alliance (HITRUST) Common Security Framework (CSF)
- Federal Risk and Authorization Management Program (FedRAMP)
- System and Organization Controls (SOC)
- Various International Organization for Standardization (ISO) certifications
- Payment Card Industry (PCI) Data Security Standard (DSS)
- The Cloud Security Alliance (CSA) Security Trust Assurance and Risk (STAR)
- United Kingdom Government Cloud (G-Cloud)
- Outsourced Service Provider's Audit Report (OSPAR)
- Korea-Information Security Management System (K-ISMS)
- Singapore Multi-Tier Cloud Security (MTCS) Level 3
- Spain Esquema Nacional de Seguridad (ENS) High-Level Security Measures

Power Virtual Agents lets you create powerful chatbots that can be created with a guided, no-code graphical interface - and without the need for data scientists or developers.

The benefits of using a no-code graphical interface help to:

- Eliminate the gap between subject matter experts and development teams building the bots

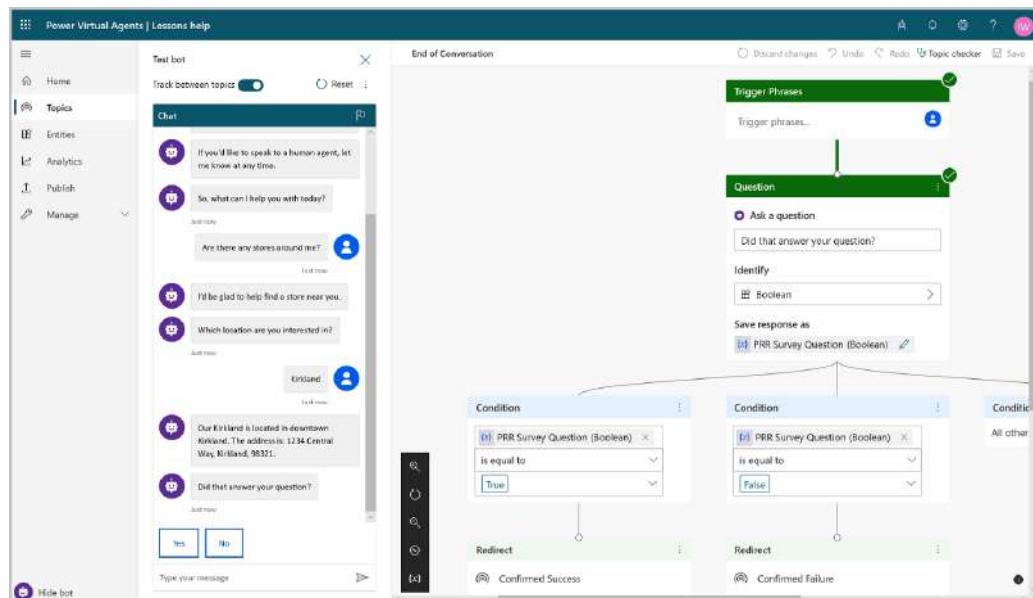
- Reduce the time from when bot builders and owners recognize an issue to when it can be updated
- Remove the need to understand complex conversational AI systems and methodologies
- Simplify the need for complex code
- Minimize the IT effort needed to deploy and maintain a custom conversational solution

Using Power Virtual Agents, you can:

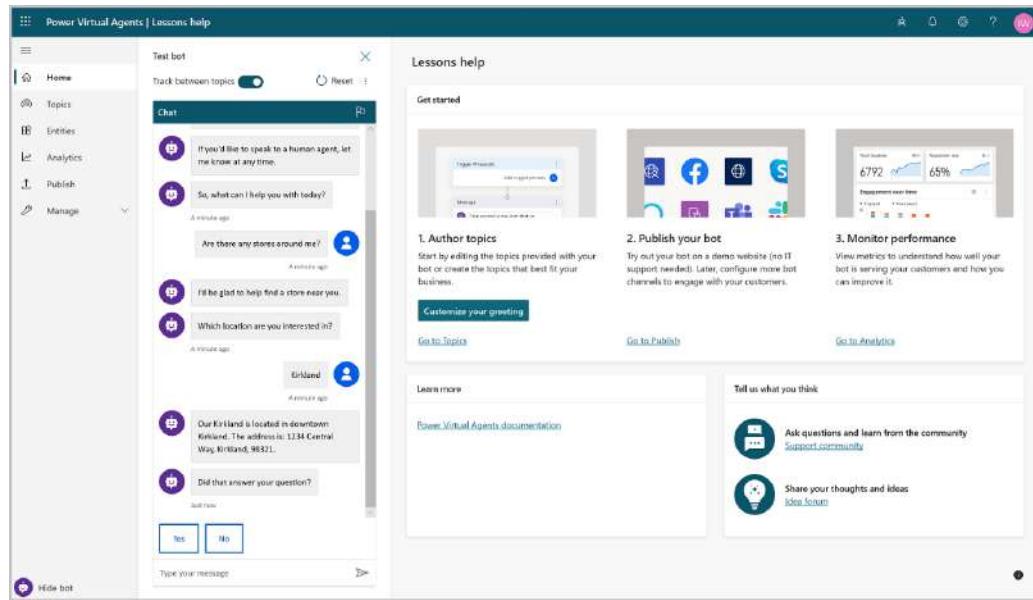
- **Empower your teams** by allowing them to easily build bots themselves without needing intermediaries, or coding or AI expertise.
- **Reduce costs** by easily automating common inquiries and freeing human agent time to deal with more complex issues.
- **Improve customer satisfaction** by allowing customers to self-help and resolve issues quickly 24/7 using rich personalized bot conversations.

## Highlights of Power Virtual Agents

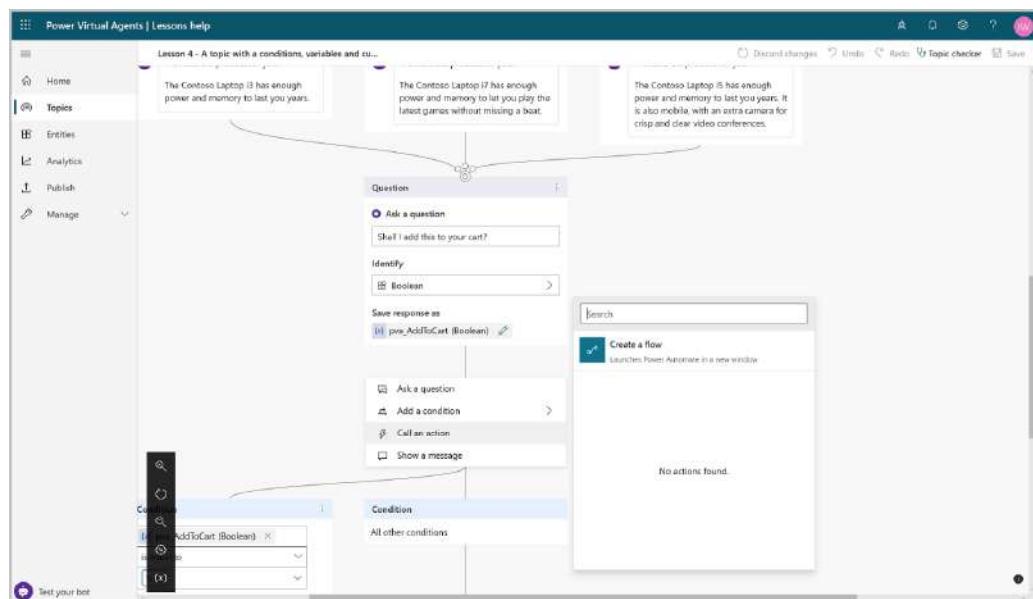
1. **Get started in seconds.** Power Virtual Agents is a software-as-a-service (SaaS) offering. It allows you to easily sign up, create your bot, and embed it into your website with just a few clicks. There's no infrastructure to maintain or complex systems to deploy.
2. **Empower your subject matter experts.** Using Power Virtual Agents, you are in the driver's seat. Your SMEs can create bots quickly and easily using a **novel, intuitive, code-free graphical interface**, eliminating the need for AI expertise or teams of developers.



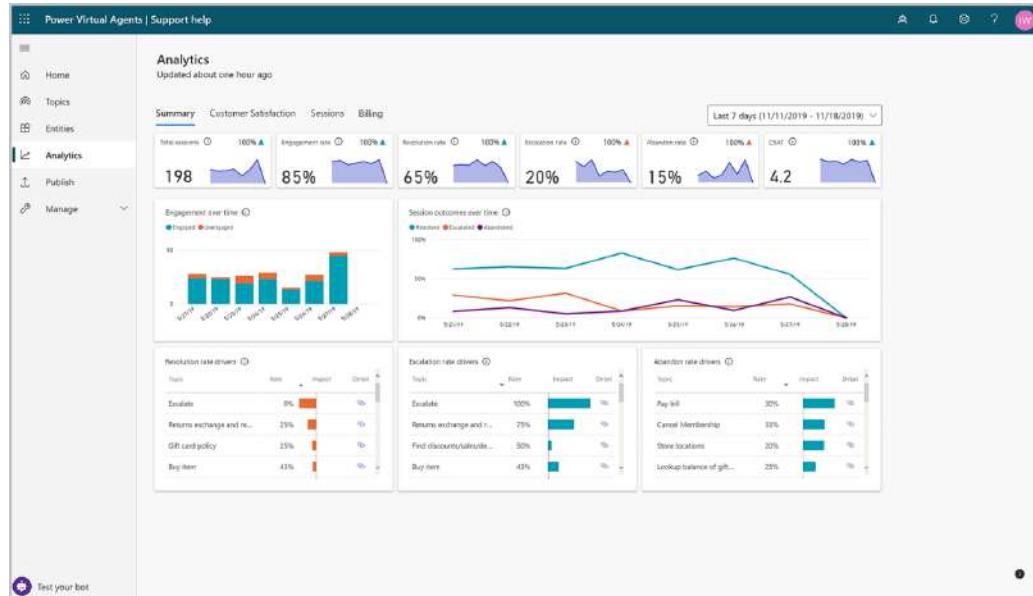
**3. Enable rich, natural conversations.** Microsoft's powerful conversational AI capabilities enable your end users to have rich multi-turn conversations that quickly guide them to the right solution. And, unlike most products on the market, there's no need to retrain AI models. Simply provide a few short examples of the topic you want the bot to handle, build the conversation using our graphical editor and your bot is ready to handle customer requests about it. You can even try out your changes in real-time in the test pane!



**4. Enable bots to take action.** Bots that can chat with your users are great, but bots that can act on their behalf are even better. With Power Virtual Agents, you can easily integrate with services and back-end systems out-of-the-box or through hundreds of easy-to-add custom connectors using Power Automate. This makes it simple to create a bot that not only responds to the user, but also takes action on their behalf.



**5. Monitor and improve bot performance.** Power Virtual Agents lets you keep an eye on how your bots are performing using powerful metrics and AI-driven dashboards. Easily see which topics are doing well and where the bot can improve, and quickly make adjustments to improve performance.



**6. Better together.** Power Virtual Agents works hand-in-hand with [Dynamics 365 Customer Service Insights](#) to provide a holistic view of your customer service operations. You can use Customer Service Insights and Power Virtual Agents together to determine which topics are trending or consuming support resources, and then easily automate them.

## Choose the right chatbot solution for your use case

A chatbot is an application that makes use of written or spoken natural language as its user interface. In other words, conversation is the means through which questions are answered, requests are serviced, and so on. Drawing upon a broad portfolio of cognitive capabilities, chatbots understand natural language and its nuances from sentence constructs to sentiment.

Chatbots can be developed as independent applications or integrated into business-application platforms.

Below you can read about the available options and when to make use of each.

**TABLE 1**

Option	Description
<a href="#">Bot Framework Composer</a>	Open-source IDE to author, test, provision, and manage chatbots.
<a href="#">Power Virtual Agents</a>	Business application platform that incorporates chatbot capability.
<a href="#">Bot Framework SDK</a>	SDK for building bots, as well as tools, templates, and related AI

**TABLE 1**

Option	Description
<a href="#">Bot Framework Orchestrator</a>	Dispatches the right skill at the right time in support of a chatbot.

## Bot Framework Composer

Bot Framework Composer is an open-source IDE for developers to author, test, provision, and manage conversational experiences. It provides a powerful visual authoring canvas for your bot logic. It lets you manage and edit:

- [Dialogs](#).
- [Language-understanding models](#).
- [Question-and-answer knowledge bases](#).
- [Bot responses](#).

With Bot Framework Composer, experiences are authored from within a single canvas. Because it is built on the [Microsoft Bot Framework](#), existing capabilities can be easily extended with code to address requirements as needed. Resulting experiences can then be tested within Composer and provisioned on Azure along with any dependent resources.

Composer is available as [a desktop application](#) for Windows, macOS, and Linux. If the desktop app is not suited to your needs, you can [build Composer from source](#) or [host Composer in the cloud](#).

As an authoring canvas, Bot Framework Composer is broadly analogous to [Power Virtual Agents](#). An important distinction, however, is that Bot Framework Composer is independent of the Microsoft Power Platform. In other words, you can make use of Bot Framework Composer to author, test, provision, and manage *standalone* conversational experiences.

## Power Virtual Agents

Power Virtual Agents is the chatbot capability included in the [Microsoft Power Platform](#)—a business-application platform that incorporates data analysis, solution building, and process automation in addition to this chatbot capability. From an easy-to-use GUI, chatbots can be built without the need to write code or to understand any of the details of the underpinning AI technologies. Because these agents can leverage automation and other capabilities within the Power Platform, sophisticated chatbot experiences can be rapidly developed. These agents can also be leveraged by [Microsoft 365](#) and [Microsoft Dynamics 365](#) for business-productivity use cases. Should there be a need, Power Virtual Agents can even tap into the SDK provided by the [Microsoft Bot Framework](#) to handle more complex scenarios.

## Bot Framework SDK

The Bot Framework, along with the Azure Bot Service, provides tools to build, test, deploy, and manage intelligent bots. The Bot Framework includes a modular and extensible SDK for building bots, as well as tools, templates, and related AI services. With this framework, developers can create bots that use speech, understand natural language, handle questions and answers, and more.

The Azure Bot Service and the Bot Framework offer:

- The Bot Framework SDK for developing bots. This SDK supports C#, Java, JavaScript, Typescript, and Python.
- Bot Framework Tools to cover end-to-end bot development workflow.
- Bot Framework Connector service to send and receive messages and events between bots and channels.
- Bot deployment and channel configuration in Azure.

Additionally, bots may make use of other Azure services, such as:

- Azure Cognitive Services to build intelligent applications.
- Azure Storage for a cloud storage solution.

Both [Power Virtual Agents](#) and [Bot Framework Composer](#) make use of the Bot Framework SDK to extend their existing capabilities and deliver more customized conversational experiences.

## Bot Framework Orchestrator

Bot Framework SDK allows to build customized discrete and reusable components of conversational logic called **skills**. Skills may be implemented as a user-facing bots or act to support another bot. As modular components, skills are advantageous for the following reasons:

- Skills help to manage complexity. Complexity is inevitable as bot-based services gain traction and use cases broaden and deepen.
- Skills promote reuse. Existing bots can be enhanced with skills; new bots can be rapidly developed.
- Skills help to ensure maintainability. Sophisticated conversational experiences may involve the involvement of multiple developers or even multiple teams.

Even though it may be composed from a number of skills, a bot must deliver a seamless experience to the end user. This is the reason for having the **Bot Framework Orchestrator**. It seamlessly dispatches the right skill at the right time in support of a conversational experience. Triggered by utterances, the Bot Framework Orchestrator dispatches the appropriate skill by recognizing the intent of the conversation and responding accordingly. Establishing the connection between an utterance and an intent requires a degree of natural language understanding and the Bot Framework Orchestrator benefits from state-of-the-art natural language understanding methods.

# Create your first bot with Composer

## Create a bot from a template

1. Open Composer.
2. Select **Create New** (+) on the homepage.
3. A list of [templates](#) is then shown which provides a starting point for your new bot. For the purposes of this quick-start, select the **Empty bot** template under the **C#** section. This template creates a bot containing a root dialog, initial greeting dialog, and an unknown intent handler. Then select **Next**.
4. Fill in the **Name** for the bot as *Menu\_bot*. Then select Azure Web App from the **Runtime type**, and a location for your bot on your machine.
5. Select **OK**. It will take a few moments for Composer to create your bot from the template.

## Add bot functionality

The bot created from the Empty bot template contains an unknown intent trigger, which acts as a fallback whenever your bot doesn't know how to respond. When your bot doesn't recognize the intent of user input, it will send a response letting the user know.

The next sections describe how to add basic bot functionality to your empty bot template. You will add:

- A [dialog](#) called **Menu** that displays a text message with menu items.
- A [trigger](#) for the **Menu** dialog. The trigger makes it possible for your bot to recognize the intent and connect to the menu dialog.

## Add a dialog

Now you can add functionality for intents that you want your bot to recognize. Dialogs are a convenient way to organize conversational logic. This section shows you how to add a dialog. In the next section, you add a trigger to the bot's main dialog to call the new dialog.

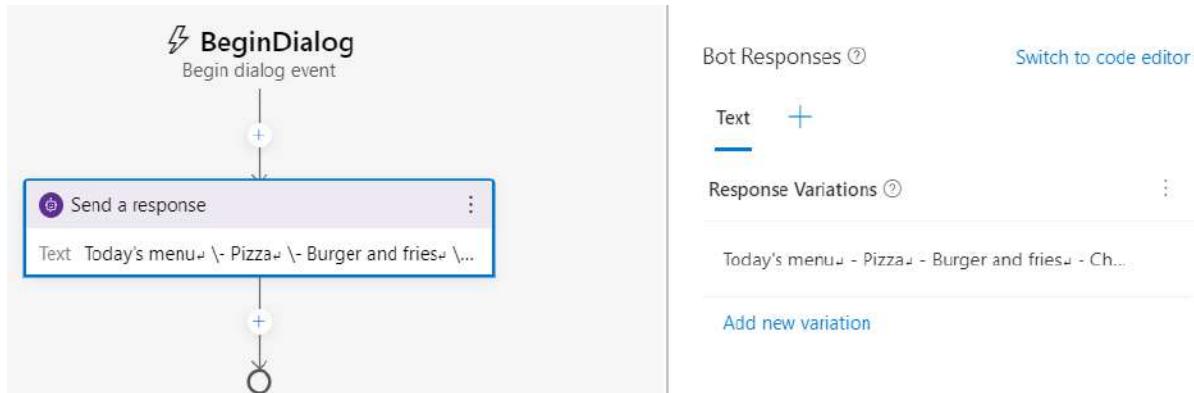
1. Click the three dots next to your bot project, **Menu\_bot**, and select **+ Add a dialog**.
2. In the **Name** field, enter *Menu*, and in the **Description** field, enter *A dialog that shows the menu..* Then select **OK**.
3. Select **BeginDialog** underneath the **Menu** dialog.
4. In the authoring canvas, select the **+** button under **BeginDialog**. Then select **Send a response**.
5. Enter the following menu text in the response editor on the right:

## Bot responseCopy

Today's menu

- Pizza
- Burger and fries
- Chicken sandwich

Your authoring canvas should look like the following:



Create a trigger to recognize the menu intent. Now that your dialog is ready to send a response of menu options, you need to create a trigger so that your bot recognizes when users want the menu item displayed.

1. Select your bot project, **Menu\_bot**.
2. On the right, select the **Regular expression recognizer** for the **Recognizer type**. This recognizer provides a simple example without adding natural language understanding. For a real-world bot, explore [language understanding](#).
3. Now, add a trigger to the **Menu\_bot** dialog.
  1. The default trigger type is **Intent recognized**.
  2. Enter **Menu** in the **What is the name of this trigger (RegEx)** field.
  3. Enter **menu** in the **Please input regEx pattern** field.
  4. Select **Submit**. Composer creates a trigger, named **Menu**, which will fire when the user sends a *menu* message to the bot.
4. Now you need to start the **Menu** dialog you created previously from the trigger. Select the **+** under the **Menu** trigger on the authoring canvas. Go to **Dialog management** and select **Begin a new dialog**.
5. Go to the right and select the box underneath **Dialog name**. You should see the **Menu** dialog you created previously; select it. Your authoring screen should look like the following:



Your menu bot is now ready to test!

### Test your bot

1. Select the **Start bot** button from the top right of Composer. The **Local runtime manager** will appear once the bot has finished building, seen below:

#### Local bot runtime manager

Start and stop local bot runtimes individually.

Bot	Status	
<input checked="" type="radio"/> weather_bot	Running	<a href="#">Open Web Chat</a> <a href="#">Test in Emulator</a>

2. Select **Open in Web Chat**. The Web Chat panel will appear on the right. Try testing different phrases. Notice that your bot will respond with the message in the **Menu** dialog if your response includes the word *menu*. Otherwise, the bot will display one of the responses from the **Unknown intent** trigger.

Congratulations! You've successfully created an echo bot!

# Create a bot with Azure

### Create an Azure Bot resource

The Azure Bot resource provides the infrastructure that allows a bot to access secured resources. It also allows the user to communicate with the bot via several channels such as Web Chat.

1. Go to the [Azure portal](#).
2. In the right pane, select **Create a resource**.
3. In the search box enter *bot*, then press **Enter**.
4. Select the **Azure Bot** card.



## Azure Bot

Microsoft

Azure Service

Build enterprise-grade conversational AI experiences with Bot Framework Composer or SDK.

Create ▾



### 5. Select **Create**.

The screenshot shows the Azure Bot service page. It features a large blue cube icon, the text "Azure Bot" in bold, and "Microsoft" below it. A "Create" button is prominently displayed at the bottom right. There are also "Add to Favorites" and rating links.

### 6. Enter the required values.

The screenshot shows the "Create an Azure Bot" wizard, step 1: Basics. It includes fields for Project details (Bot handle, Subscription, Resource group), Pricing (Pricing tier: Standard), Microsoft App ID (Create new Microsoft App ID selected), and a note about app secret storage in Azure Key Vault. Navigation buttons at the bottom include "Review + create" and "Next : Tags >".

Basics   Tags   Review + create

**Project details**  
Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Bot handle \*

Subscription \*

Resource group \*  Create new

**Pricing**  
Select a pricing tier for your Azure Bot resource. You can change your selection later in the Azure portal's resource management. Learn more about available options, or request a pricing quote, by visiting the [Azure Bot Services pricing](#).

Pricing tier

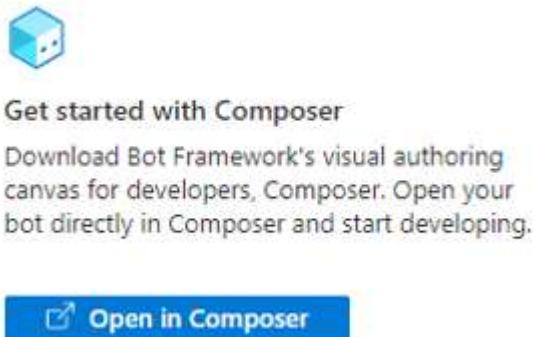
**Microsoft App ID**  
A Microsoft App ID is required to create an Azure Bot resource. An App ID can be automatically created below, or you can manually create your own, then return here to input your new App ID and password.  
[Manually create App ID](#)

The app secret will be stored in Azure Key Vault in the same resource group as your Azure bot. [Learn More](#)

Microsoft App ID  Create new Microsoft App ID  Use existing app registration

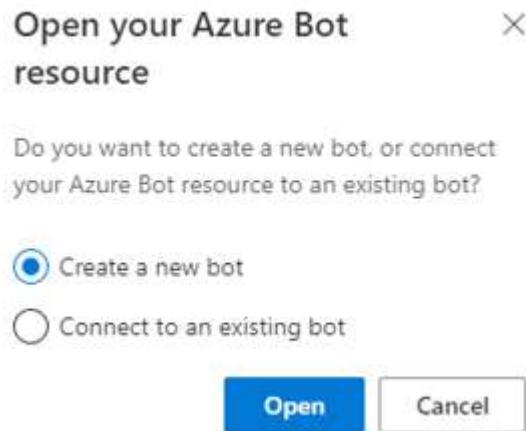
[Review + create](#) [Previous](#) [Next : Tags >](#)

7. Select **Review + create**.
8. If the validation passes, select **Create**. You should see the **Azure Bot** and the related key vault resources listed in the resource group you selected.
9. Select **Open in Composer**.



The Composer application opens. If the application isn't installed, you will be asked to install it before you can proceed with the next steps.

1. In the pop-up window, select **Create a new bot**.



## Azure Key Vault

When Azure creates the Azure Bot resource, it also generates an **app ID** and an **app password** and stores the password in Azure Key Vault. Key Vault is a service that provides centralized secrets management, with full control over access policies and audit history. For more information, see [Use Key Vault references for App Service and Azure Functions](#). You're charged a small fee for using the service.

## Create a bot

Create a bot by following the steps described below.

Create a bot from a template:

1. Open Composer.
2. Select **Create New** (+) on the homepage.
3. Composer shows a list of [templates](#). The templates provide a starting point for your new bot. For the purposes of this quick-start, select the **Empty bot** template under the **C#** section. This template creates a bot containing a root dialog, initial greeting dialog, and an unknown intent handler. Then select **Next**.
4. Fill in the **Name** for the bot as *Menu\_bot*. Then select Azure Web App from the **Runtime type**, and a location for your bot on your machine.
5. Select **OK**. It will take a few moments for Composer to create your bot from the template.

## Add bot functionality

The bot created from the Empty bot template contains an unknown intent trigger. This acts as a fallback whenever your bot doesn't know how to respond. When your bot doesn't recognize the intent of user input, it will send a response letting the user know.

This section describes how to add basic bot functionality to your empty bot template. You will add:

- A [dialog](#) called **Menu** that displays a text message with menu items.
- A [trigger](#) for the **Menu** dialog. This makes it possible for your bot to recognize the intent, and connects the menu dialog above.

## Add a dialog

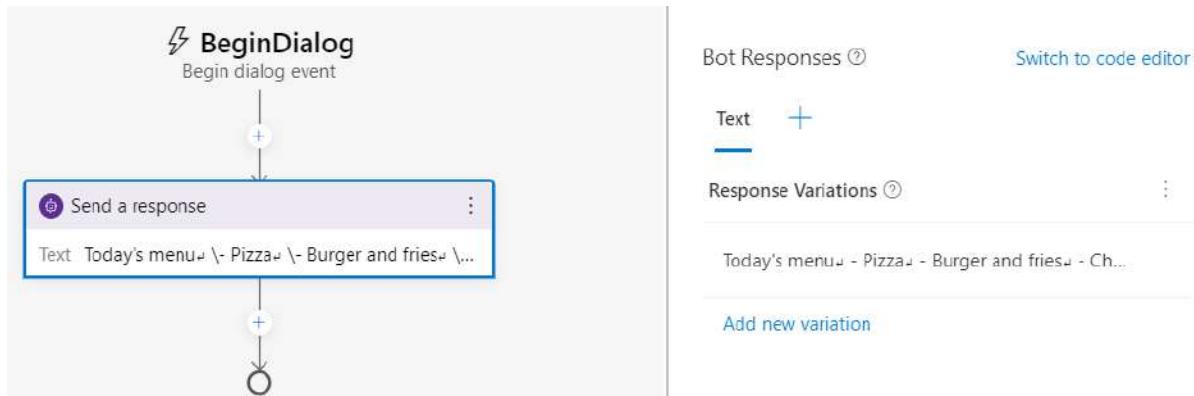
Now you can add functionality for intents that you want your bot to recognize. Dialogs are a convenient way to organize conversational logic. This section shows you how to add a dialog. In the next section, you add a trigger to the bot's main dialog to call the new dialog.

1. Click the three dots next to your bot project, **Menu\_bot**, and select **+ Add a dialog**.
2. In the **Name** field, enter *Menu*, and in the **Description** field, enter *A dialog that shows the menu..* Then select **OK**.
3. Select **BeginDialog** underneath the **Menu** dialog.
4. In the authoring canvas, select the + button under **BeginDialog**. Then select **Send a response**.
5. Enter the following menu text in the response editor on the right.

## Bot responseCopy

```
Today's menu
- Pizza
- Burger and fries
- Chicken sandwich
```

Your authoring canvas should look like the following.



*Create a trigger to recognize the menu intent*

Now that your dialog is ready to send a response of menu options, you need to create a trigger so that your bot recognizes when users want the menu item displayed.

1. Select your bot project, **Menu\_bot**.
2. On the right, select the **Regular expression recognizer** for the **Recognizer type**. This recognizer provides a simple example without adding natural language understanding
3. Now, add a trigger to the **Menu\_bot** dialog.
  1. The default trigger type is **Intent recognized**.
  2. Enter **Menu** in the **What is the name of this trigger (RegEx)** field.
  3. Enter **menu** in the **Please input regEx pattern** field.
  4. Select **Submit**. This creates a trigger, named **Menu**, which will fire when the user sends a *menu* message to the bot.
4. Now you need to start the **Menu** dialog you created previously from the trigger. Select the **+** under the **Menu** trigger on the authoring canvas. Go to **Dialog management** and select **Begin a new dialog**.
5. Go to the right and select the box underneath **Dialog name**. You should see the **Menu** dialog you created previously; select it. Your authoring canvas should look like the following.



Your menu bot is now ready to test!

Next step:

Publish a bot to azure

## Azure Cognitive Search

### What is Azure Cognitive Search?

Azure Cognitive Search ([formerly known as "Azure Search"](#)) is a cloud search service that gives developers infrastructure, APIs, and tools for building a rich search experience over private, heterogeneous content in web, mobile, and enterprise applications.

Search is foundational to any app that surfaces text content to users, with common scenarios including catalog or document search, online retail, or knowledge mining for data science. When you create a search service, you'll work with the following capabilities:

- A search engine for full text search with storage for user-owned content in a search index
- Rich indexing, with [text analysis](#) and [optional AI enrichment](#) for advanced content extraction and transformation
- Rich query capabilities, including simple syntax, full Lucene syntax, and typeahead search
- Programmability through REST APIs and client libraries in Azure SDKs for .NET, Python, Java, and JavaScript
- Azure integration at the data layer, machine learning layer, and AI (Cognitive Services)

Architecturally, a search service sits between the external data stores that contain your un-indexed data, and your client app that sends query requests to a search index and handles the response.

Across the Azure platform, Cognitive Search can integrate with other Azure services in the form of *indexers* that automate data ingestion/retrieval from Azure data sources, and *skillsets* that incorporate consumable AI from Cognitive Services, such as image and

natural language processing, or custom AI that you create in Azure Machine Learning or wrap inside Azure Functions.

## Inside a search service

On the search service itself, the two primary workloads are *indexing* and *querying*.

- [Indexing](#) is an intake process that loads content into to your search service and makes it searchable. Internally, inbound text is processed into tokens and stored in inverted indexes for fast scans. You can upload any text that is in the form of JSON documents.

Additionally, if your content includes mixed files, you have the option of adding *AI enrichment* through [cognitive skills](#). AI enrichment can extract text embedded in application files, and also infer text and structure from non-text files by analyzing the content.

The skills providing the analysis are predefined ones from Microsoft, or custom skills that you create. The subsequent analysis and transformations can result in new information and structures that did not previously exist, providing high utility for many search and knowledge mining scenarios.

- [Querying](#) can happen once an index is populated with searchable text, when your client app sends query requests to a search service and handles responses. All query execution is over a search index that you create, own, and store in your service. In your client app, the search experience is defined using APIs from Azure Cognitive Search, and can include relevance tuning, autocomplete, synonym matching, fuzzy matching, pattern matching, filter, and sort.

Functionality is exposed through a simple [REST API](#) or [.NET SDK](#) that masks the inherent complexity of information retrieval. You can also use the Azure portal for service administration and content management, with tools for prototyping and querying your indexes and skillsets. Because the service runs in the cloud, infrastructure and availability are managed by Microsoft.

## Why use Cognitive Search?

Azure Cognitive Search is well suited for the following application scenarios:

- Consolidate heterogeneous content into a private, user-defined search index. Offload indexing and query workloads onto a dedicated search service.
- Easily implement search-related features: relevance tuning, faceted navigation, filters (including geo-spatial search), synonym mapping, and autocomplete.

- Transform large undifferentiated text or image files, or application files stored in Azure Blob Storage or Cosmos DB, into searchable JSON documents. This is achieved during index through [cognitive skills](#) that add external processing.
- Add linguistic or custom text analysis. If you have non-English content, Azure Cognitive Search supports both Lucene analyzers and Microsoft's natural language processors. You can also configure analyzers to achieve specialized processing of raw content, such as filtering out diacritics, or recognizing and preserving patterns in strings.

## How to get started

An end-to-end exploration of core search features can be achieved in four steps:

1. **Create a search service** at the shared Free tier or a [billable tier](#) for dedicated resources used only by your service. All quickstarts and tutorials can be completed on a shared service.
2. **Create a search index** using the portal, [REST API](#), [.NET SDK](#), or another SDK. The index schema defines the structure of searchable content.
3. **Upload content** using the "push" model to push JSON documents from any source, or use the "pull" model ([indexers](#)) if your source data is on Azure.
4. **Query an index** using [Search explorer](#) in the portal, [REST API](#), [.NET SDK](#), or another SDK.

For initial exploration, start with the [Import data wizard](#) and a built-in Azure data source to create, load, and query an index in minutes.

## Compare search options

Customers often ask how Azure Cognitive Search compares with other search-related solutions. The following table summarizes key differences.

## COMPARE SEARCH OPTIONS

Compared to	Key differences
Microsoft Search	<p><a href="#">Microsoft Search</a> is for Microsoft 365 authenticated users who need to query over content in SharePoint. It's offered as a ready-to-use search experience, enabled and configured by administrators, with the ability to accept external content through connectors from Microsoft and other sources. If this describes your scenario, then Microsoft Search with Microsoft 365 is an attractive option to explore.</p> <p>In contrast, Azure Cognitive Search executes queries over an index that you define, populated with data and documents you own, often from diverse sources. Azure Cognitive Search has crawler capabilities for some Azure data sources through <a href="#">indexers</a>, but you can push any JSON document that conforms to your index schema into a single, consolidated searchable resource. You can also customize the indexing pipeline to include machine learning and lexical analyzers. Because Cognitive Search is built to be a plug-in component in larger solutions, you can integrate search</p>
Bing	<p><a href="#">Bing Web Search API</a> searches the indexes on Bing.com for matching terms you submit. Indexes are built from HTML, XML, and other web content on public sites. Built on the same foundation, <a href="#">Bing Custom Search</a> offers the same crawler technology for web content types, scoped to individual web sites.</p> <p>In Cognitive Search, you can define and populate the index. You can use <a href="#">indexers</a> to crawl data on Azure data sources, or push any index-conforming JSON document to your search service.</p>
Database search	<p>Many database platforms include a built-in search experience. SQL Server has <a href="#">full text search</a>. Cosmos DB and similar technologies have queryable indexes. When evaluating products that combine search and storage, it can be challenging to determine which way to go. Many solutions use both: DBMS for storage, and Azure Cognitive Search for specialized search features.</p> <p>Compared to DBMS search, Azure Cognitive Search stores content from heterogeneous sources and offers specialized text processing features such as linguistic-aware text processing (stemming, lemmatization, word forms) in <a href="#">56 languages</a>. It also supports autocorrection of misspelled words, <a href="#">synonyms</a>, <a href="#">suggestions</a>, <a href="#">scoring controls</a>, <a href="#">facets</a>, and <a href="#">custom tokenization</a>. The <a href="#">full text search engine</a> in Azure Cognitive Search is built on Apache Lucene, an industry standard in information retrieval. However, while Azure Cognitive Search persists data in the form of an inverted index, it is not a replacement for true data storage and we don't recommend using it in that capacity. For more information, see this <a href="#">forum post</a>.</p> <p>Resource utilization is another inflection point in this category. Indexing and some query operations are often computationally intensive. Offloading search from the DBMS to a dedicated solution in the cloud preserves system resources for transaction</p>

## COMPARE SEARCH OPTIONS

Compared to	Key differences
Dedicated search solution	<p>Assuming you have decided on dedicated search with full spectrum functionality, a final categorical comparison is between on premises solutions or a cloud service. Many search technologies offer controls over indexing and query pipelines, access to richer query and filtering syntax, control over rank and relevance, and features for self-directed and intelligent search.</p> <p>A cloud service is the right choice if you want a turn-key solution with minimal overhead and maintenance, and adjustable scale.</p> <p>Within the cloud paradigm, several providers offer comparable baseline features, with full-text search, geo-search, and the ability to handle a certain level of ambiguity in search inputs. Typically, it's a <a href="#">specialized feature</a>, or the ease and overall simplicity of APIs, tools, and management that determines the best fit.</p>

Among cloud providers, Azure Cognitive Search is strongest for full text search workloads over content stores and databases on Azure, for apps that rely primarily on search for both information retrieval and content navigation.

Key strengths include:

- Azure data integration (crawlers) at the indexing layer
- Azure Private Link integration to support off-internet security requirements
- Integration with AI processing to make unsearchable content types text-searchable.
- Linguistic and custom analysis, with analyzers for solid full text search in 56 languages
- [Critical features](#): rich query language, relevance tuning, faceting, autocomplete, synonyms, geo-search, and result composition.
- Azure scale, reliability, and world-class availability

Among our customers, those able to leverage the widest range of features in Azure Cognitive Search include online catalogs, line-of-business programs, and document discovery applications.

# Features of Azure Cognitive Search

Azure Cognitive Search provides a full-text search engine, persistent storage of search indexes, integrated AI used during indexing to extract more text and structure, and APIs and tools.

The following table summarizes features by category. For more information about how Cognitive Search compares with other search technologies.

## Indexing features

INDEXING FEATURES	
Category	Features
Data sources	<p>Search indexes can accept text from any source, provided it is submitted as a JSON document.</p> <p><b>Indexers</b> are a feature that automates data import from supported data sources to extract searchable content in primary data stores. Indexers handle JSON serialization for you. You can connect to a <a href="#">variety of data sources</a>, including <a href="#">Azure SQL Database</a>, <a href="#">Azure Cosmos DB</a>, or <a href="#">Azure Blob</a></p>
Hierarchical and nested data structures	<p><b>Complex types</b> and collections allow you to model virtually any type of JSON structure within a search index. One-to-many and many-to-many cardinality can be expressed natively through collections, complex types, and collections of complex types.</p>
Linguistic analysis	<p>Analyzers are components used for text processing during indexing and search operations. By default, you can use the general-purpose Standard Lucene analyzer, or override the default with a language analyzer, a custom analyzer that you configure, or another predefined analyzer that produces tokens in the format you require.</p> <p><b>Language analyzers</b> from Lucene or Microsoft are used to intelligently handle language-specific linguistics including verb tenses, gender, irregular plural nouns (for example, 'mouse' vs. 'mice'), word de-compounding, word-breaking (for languages with no spaces), and more.</p> <p><b>Custom lexical analyzers</b> are used for complex query forms such as phonetic matching and regular expressions.</p>

## AI enrichment and knowledge mining

## AI ENRICHMENT AND KNOWLEDGE MINING

Category	Features
AI processing during indexing	<b>AI enrichment</b> refers to embedded image and natural language processing in an indexer pipeline that extracts text and information from content that cannot otherwise be indexed for full text search. AI processing is achieved by adding and combining skills in a skillset, which is then attached to an indexer. AI can be either <a href="#">built-in skills</a> from Microsoft, such as text translation or Optical Character Recognition (OCR), or <a href="#">custom skills</a> that you provide.
Storing enriched content for analysis and consumption in non-search scenarios	<b>Knowledge store</b> is persistent storage of enriched content, intended for non-search scenarios like knowledge mining and data science processing. A knowledge store is defined in a skillset, but created in Azure Storage as objects or tabular rowsets.
Cached enrichments	<b>Incremental enrichment (preview)</b> refers to cached enrichments that can be reused during skillset execution. Caching is particularly valuable in skillsets that include OCR and image analysis, which are expensive to process.

Query and user experience

## QUERY AND USER EXPERIENCE

Category	Features
Free-form text search	<p><b>Full-text search</b> is a primary use case for most search-based apps. Queries can be formulated using a supported syntax.</p> <p><b>Simple query syntax</b> provides logical operators, phrase search operators, suffix operators, precedence operators.</p> <p><b>Full Lucene query syntax</b> includes all operations in simple syntax, with extensions for fuzzy search, proximity search, term boosting, and regular expressions.</p>
Relevance	<b>Simple scoring</b> is a key benefit of Azure Cognitive Search. Scoring profiles are used to model relevance as a function of values in the documents themselves. For example, you might want newer products or discounted products to appear higher in the search results. You can also build scoring profiles using tags for personalized scoring based on customer search preferences you've tracked and stored separately.
Geo-search	Azure Cognitive Search processes, filters, and displays geographic locations. It enables users to explore data based on the proximity of a search result to a physical location.
Filters and facets	<p><b>Faceted navigation</b> is enabled through a single query parameter. Azure Cognitive Search returns a faceted navigation structure you can use as the code behind a categories list, for self-directed filtering (for example, to filter catalog items by price-range or brand).</p> <p><b>Filters</b> can be used to incorporate faceted navigation into your application's UI, enhance query formulation, and filter based on user- or developer-specified criteria. Create filters using the OData syntax.</p>
User experience	<p><b>Autocomplete</b> can be enabled for type-ahead queries in a search bar.</p> <p><b>Search suggestions</b> also works off of partial text inputs in a search bar, but the results are actual documents in your index rather than query terms.</p> <p><b>Synonyms</b> associates equivalent terms that implicitly expand the scope of a query, without the user having to provide the alternate terms.</p> <p><b>Hit highlighting</b> applies text formatting to a matching keyword in search results. You can choose which fields return highlighted snippets.</p> <p><b>Sorting</b> is offered for multiple fields via the index schema and then toggled at query-time with a single search parameter.</p> <p><b>Paging</b> and throttling your search results is straightforward with the finely tuned control that Azure Cognitive Search offers over your search results.</p>

## Security features

SECURITY FEATURES	
Category	Features
Data encryption	<p><b>Microsoft-managed encryption-at-rest</b> is built into the internal storage layer and is irrevocable.</p> <p><b>Customer-managed encryption keys</b> that you create and manage in Azure Key Vault can be used for supplemental encryption of indexes and synonym maps. For services created after August 1 2020, CMK encryption extends to data on temporary disks, for full double encryption of indexed</p>
Endpoint protection	<p><b>IP rules for inbound firewall support</b> allows you to set up IP ranges over which the search service will accept requests.</p> <p><b>Create a private endpoint</b> using Azure Private Link to force all requests through a virtual network.</p>
Outbound security (indexers)	<p><b>Data access through private endpoints</b> allows an indexer to connect to Azure resources that are protected through Azure Private Link.</p> <p><b>Data access using a trusted identity</b> means that connection strings to external data sources can omit user names and passwords. When an indexer connects to the data source, the resource allows the connection if the search service was previously registered as a trusted service.</p>

## Portal features

PORTAL FEATURES	
Category	Features
Tools for prototyping and inspection	<p><b>Add index</b> is an index designer in the portal that you can use to create a basic schema consisting of attributed fields and a few other settings. After saving the index, you can populate it using an SDK or the REST API to provide the data.</p> <p><b>Import data wizard</b> creates indexes, indexers, skillsets, and data source definitions. If your data exists in Azure, this wizard can save you significant time and effort, especially for proof-of-concept investigation and exploration.</p> <p><b>Search explorer</b> is used to test queries and refine scoring profiles.</p> <p><b>Create demo app</b> is used to generate an HTML page that can be used to test the search experience.</p>
Monitoring and diagnostics	<b>Enable monitoring features</b> to go beyond the metrics-at-a-glance that are always visible in the portal. Metrics on queries per second, latency, and throttling are captured and reported in portal pages with no additional configuration required.

## Programmability

### PROGRAMMABILITY

Category	Features
REST	<p><a href="#">Service REST API</a> is for data plane operations, including all operations related to indexing, queries, and AI enrichment. You can also use this client library to retrieve system information and statistics.</p> <p><a href="#">Management REST API</a> is for service creation and clean up through Azure Resource Manager. You can also use this API to manage keys and provision a service.</p>
Azure SDK for .NET	<p><a href="#">Azure.Search.Documents</a> is for data plane operations, including all operations related to indexing, queries, and AI enrichment. You can also use this client library to retrieve system information and statistics.</p> <p><a href="#">Microsoft.Azure.Management.Search</a> is for service creation and clean up through Azure Resource Manager. You can also use this API to manage keys and provision a service.</p>
Azure SDK for Java	<p><a href="#">com.azure.search.documents</a> is for data plane operations, including all operations related to indexing, queries, and AI enrichment. You can also use this client library to retrieve system information and statistics.</p> <p><a href="#">com.microsoft.azure.management.search</a> is for service creation and clean up through Azure Resource Manager. You can also use this API to manage keys and provision a service.</p>
Azure SDK for Python	<p><a href="#">azure-search-documents</a> is for data plane operations, including all operations related to indexing, queries, and AI enrichment. You can also use this client library to retrieve system information and statistics.</p> <p><a href="#">azure-mgmt-search</a> is for service creation and clean up through Azure Resource Manager. You can also use this API to manage keys and provision a service.</p>
Azure SDK for JavaScript/TypeScript	<p><a href="#">azure/search-documents</a> is for data plane operations, including all operations related to indexing, queries, and AI enrichment. You can also use this client library to retrieve system information and statistics.</p> <p><a href="#">azure/arm-search</a> is for service creation and clean up through Azure Resource Manager. You can also use this API to manage keys and provision a service.</p>

# Azure Cognitive Services

## What are Azure Cognitive Services?

Azure Cognitive Services are cloud-based services with REST APIs and client library SDKs available to help you build cognitive intelligence into your applications. You can add cognitive features to your applications without having artificial intelligence (AI) or data science skills. Azure Cognitive Services comprise various AI services that enable you to build cognitive solutions that can see, hear, speak, understand, and even make decisions.

### Categories of Cognitive Services

The catalog of cognitive services that provide cognitive understanding is categorized into five main pillars:

- Vision
- Speech
- Language
- Decision
- Search

The following sections in this article provide a list of services that are part of these five pillars.

## Vision APIs

### VISION APIs

Service Name	Service Description
Computer Vision	The Computer Vision service provides you with access to advanced cognitive algorithms for processing images and returning information. See <a href="#">Computer Vision quickstart</a> to get started with the service.
Custom Vision Service	The Custom Vision Service lets you build, deploy, and improve your own image classifiers. An image classifier is an AI service that applies labels to images, based on their visual characteristics.
Face	The Face service provides access to advanced face algorithms, enabling face attribute detection and recognition. See <a href="#">Face quickstart</a> to get started with the

## Speech APIs

### SPEECH APIs

Service Name	Service Description
Speech service	Speech service adds speech-enabled features to applications. Speech service includes various capabilities like speech-to-text, text-to-speech, speech translation, and many

## Language APIs

### LANGUAGE APIs

Service Name	Service Description
Azure Cognitive Service for language	Azure Cognitive Service for Language provides several Natural Language Processing (NLP) features for understanding and analyzing text.
Language Understanding LUIS	Language Understanding (LUIS) is a cloud-based conversational AI service that applies custom machine-learning intelligence to a user's conversational, natural language text to predict overall meaning, and pull out relevant, detailed information. See <a href="#">LUIS quickstart</a> to get started with the service.
QnA Maker	QnA Maker allows you to build a question and answer service from your semi-structured content. See <a href="#">QnA Maker quickstart</a> to get started with the service.
Translator	Translator provides machine-based text translation in near real-time.

## Decision APIs

DECISION APIs	
Service	Service Description
Anomaly Detector	Anomaly Detector allows you to monitor and detect abnormalities in your time series data. See <a href="#">Anomaly Detector quickstart</a> to get started with the service.
Content Moderator	Content Moderator provides monitoring for possible offensive, undesirable, and risky content. See <a href="#">Content Moderator quickstart</a> to get started with the service.
Personalizer	Personalizer allows you to choose the best experience to show to your users, learning from their real-time behavior. See <a href="#">Personalizer quickstart</a> to get started

## Search APIs

SEARCH APIs	
Service Name	Service Description
Bing News Search	Bing News Search returns a list of news articles determined to be relevant to the user's query.
Bing Video Search	Bing Video Search returns a list of videos determined to be relevant to the user's query.
Bing Web Search	Bing Web Search returns a list of search results determined to be relevant to the user's query.
Bing Autosuggest	Bing Autosuggest allows you to send a partial search query term to Bing and get back a list of suggested queries.
Bing Custom Search	Bing Custom Search allows you to create tailored search experiences for topics that you care about.
Bing Entity Search	Bing Entity Search returns information about entities that Bing determines are relevant to a user's query.
Bing Image Search	Bing Image Search returns a display of images determined to be relevant to the user's query.
Bing Visual Search	Bing Visual Search returns insights about an image such as visually similar images, shopping sources for products found in the image, and related
Bing Local Business Search	Bing Local Business Search API enables your applications to find contact and location information about local businesses based on search queries.
Bing Spell Check	Bing Spell Check allows you to perform contextual grammar and spell checking.

## Get started with Cognitive Services

Start by creating a Cognitive Services resource with hands-on quickstarts using the following methods:

- [Azure portal](#)
- [Azure CLI](#)
- [Azure SDK client libraries](#)
- [Azure Resource Manager \(ARM\) templates](#)

## Using Cognitive Services in different development environments

With Azure and Cognitive Services, you have access to several development options, such as:

- Automation and integration tools like Logic Apps and Power Automate.
- Deployment options such as Azure Functions and the App Service.
- Cognitive Services Docker containers for secure access.
- Tools like Apache Spark, Azure Databricks, Azure Synapse Analytics, and Azure Kubernetes Service for Big Data scenarios.

## Using Cognitive Services securely

Azure Cognitive Services provides a layered security model, including [authentication](#) via Azure Active Directory credentials, a valid resource key, and [Azure Virtual Networks](#).

## Containers for Cognitive Services

Azure Cognitive Services provides several Docker containers that let you use the same APIs that are available in Azure, on-premises. Using these containers gives you the flexibility to bring Cognitive Services closer to your data for compliance, security or other operational reasons

## Regional availability

The APIs in Cognitive Services are hosted on a growing network of Microsoft-managed data centers. You can find the regional availability for each API in [Azure region list](#).

## supported cultural languages

Cognitive Services supports a wide range of cultural languages at the service level. You can find the language availability for each API in the [supported languages list](#).

# Natural language support for Azure Cognitive Services

Azure Cognitive Services enable you to build applications that see, hear, speak with, and understand your users. Between these services, more than three dozen languages are supported, allowing users to communicate with your application in natural ways. Use the links below to view language availability by service.

These Cognitive Services are language agnostic and don't have limitations based on human language.

- [Anomaly Detector \(Preview\)](#)
- [Custom Vision](#)
- [Face](#)
- [Personalizer](#)

## Vision

- [Computer Vision](#)
- [Ink Recognizer \(Preview\)](#)
- [Video Indexer](#)

## Language

- [Language Understanding \(LUIS\)](#)
- [QnA Maker](#)
- [Language service](#)
- [Translator](#)

## Speech

- [Speech Service: Speech-to-Text](#)
- [Speech Service:Text-to-Speech](#)
- [Speech Service: Speech Translation](#)

## Decision

- [Content Moderator](#)

# Cognitive Services development options

Azure Cognitive Services are cloud-based AI services that allow developers to build intelligence into their applications and products without deep knowledge of machine learning. With Cognitive Services, you have access to AI capabilities or models that are built, trained, and updated by Microsoft - ready to be used in your applications. In many cases, you also have the option to customize the models for your business needs.

Cognitive Services are organized into four categories: Decision, Language, Speech, and Vision. Typically you would access these services through REST APIs, client libraries, and custom tools (like command-line interfaces) provided by Microsoft. However, this is only one path to success. Through Azure, you also have access to several development options, such as:

- Automation and integration tools like Logic Apps and Power Automate.
- Deployment options such as Azure Functions and the App Service.
- Cognitive Services Docker containers for secure access.
- Tools like Apache Spark, Azure Databricks, Azure Synapse Analytics, and Azure Kubernetes Service for Big Data scenarios.

Before we jump in, it's important to know that the Cognitive Services are primarily used for two distinct tasks. Based on the task you want to perform, you have different development and deployment options to choose from.

- [Development options for prediction and analysis](#)
- [Tools to customize and configure models](#)

## Development options for prediction and analysis

The tools that you will use to customize and configure models are different from those that you'll use to call the Cognitive Services. Out of the box, most Cognitive Services allow you to send data and receive insights without any customization. For example:

- You can send an image to the Computer Vision service to detect words and phrases or count the number of people in the frame
- You can send an audio file to the Speech service and get transcriptions and translate the speech to text at the same time

Azure offers a wide range of tools that are designed for different types of users, many of which can be used with Cognitive Services. Designer-driven tools are the easiest to use, and are quick to set up and automate, but may have limitations when it comes to customization. Our REST APIs and client libraries provide users with more control and flexibility, but require more effort, time, and expertise to build a solution. If you use REST APIs and client libraries, there is an expectation that you're comfortable working with modern programming languages like C#, Java, Python, JavaScript, or another popular programming language.

Let's take a look at the different ways that you can work with the Cognitive Services.

## Client libraries and REST APIs

Cognitive Services client libraries and REST APIs provide you direct access to your service. These tools provide programmatic access to the Cognitive Services, their baseline models, and in many cases allow you to programmatically customize your models and solutions.

- **Target user(s):** Developers and data scientists
- **Benefits:** Provides the greatest flexibility to call the services from any language and environment.
- **UI:** N/A - Code only
- **Subscription(s):** Azure account + Cognitive Services resources

If you want to learn more about available client libraries and REST APIs, use our [Cognitive Services overview](#) to pick a service and get started with one of our quickstarts for vision, decision, language, and speech.

## Cognitive Services for Big Data

With Cognitive Services for Big Data you can embed continuously improving, intelligent models directly into Apache Spark™ and SQL computations. These tools liberate developers from low-level networking details, so that they can focus on creating smart, distributed applications. Cognitive Services for Big Data support the following platforms and connectors: Azure Databricks, Azure Synapse, Azure Kubernetes Service, and Data Connectors.

- **Target user(s):** Data scientists and data engineers
- **Benefits:** The Azure Cognitive Services for Big Data let users channel terabytes of data through Cognitive Services using Apache Spark™. It's easy to create large-scale intelligent applications with any datastore.
- **UI:** N/A - Code only
- **Subscription(s):** Azure account + Cognitive Services resources

If you want to learn more about Big Data for Cognitive Services, a good place to start is with the [overview](#). If you're ready to start building, try our [Python](#) or [Scala](#) samples.

## Azure Functions and Azure Service Web Jobs

[Azure Functions](#) and [Azure App Service Web Jobs](#) both provide code-first integration services designed for developers and are built on [Azure App Services](#). These products provide serverless infrastructure for writing code. Within that code you can make calls to our services using our client libraries and REST APIs.

- **Target user(s):** Developers and data scientists
- **Benefits:** Serverless compute service that lets you run event-triggered code.
- **Subscription(s):** Azure account + Cognitive Services resource + Azure Functions subscription

## Azure Logic Apps

[Azure Logic Apps](#) share the same workflow designer and connectors as Power Automate but provide more advanced control, including integrations with Visual Studio and DevOps. Power Automate makes it easy to integrate with your Cognitive Services resources through service-specific connectors that provide a proxy or wrapper around the APIs. These are the same connectors as those available in Power Automate.

- **Target user(s):** Developers, integrators, IT pros, DevOps
- **Benefits:** Designer-first (declarative) development model providing advanced options and integration in a low-code solution
- **UI:** Yes
- **Subscription(s):** Azure account + Cognitive Services resource + Logic Apps deployment

## Power Automate

Power Automate is a service in the [Power Platform](#) that helps you create automated workflows between apps and services without writing code. We offer several connectors to make it easy to interact with your Cognitive Services resource in a Power Automate solution. Power Automate is built on top of Logic Apps.

- **Target user(s):** Business users (analysts) and SharePoint administrators
- **Benefits:** Automate repetitive manual tasks simply by recording mouse clicks, keystrokes and copy paste steps from your desktop!
- **UI tools:** Yes - UI only
- **Subscription(s):** Azure account + Cognitive Services resource + Power Automate Subscription + Office 365 Subscription

## AI Builder

[AI Builder](#) is a Microsoft Power Platform capability you can use to improve business performance by automating processes and predicting outcomes. AI Builder brings the power of AI to your solutions through a point-and-click experience. Many cognitive services such as the Language service, and Computer Vision have been directly integrated here and you don't need to create your own Cognitive Services.

- **Target user(s):** Business users (analysts) and SharePoint administrators
- **Benefits:** A turnkey solution that brings the power of AI through a point-and-click experience. No coding or data science skills required.
- **UI tools:** Yes - UI only
- **Subscription(s):** AI Builder

## Continuous integration and deployment

You can use Azure DevOps and GitHub actions to manage your deployments. In the [section below](#), we have two examples of CI/CD integrations to train and deploy custom models for Speech and the Language Understanding (LUIS) service.

- **Target user(s):** Developers, data scientists, and data engineers
- **Benefits:** Allows you to continuously adjust, update, and deploy applications and models programmatically. There is significant benefit when regularly using your data to improve and update models for Speech, Vision, Language, and Decision.
- **UI tools:** N/A - Code only
- **Subscription(s):** Azure account + Cognitive Services resource + GitHub account

# Cognitive Services and machine learning

Cognitive Services provides machine learning capabilities to solve general problems such as analyzing text for emotional sentiment or analyzing images to recognize objects or faces. You don't need special machine learning or data science knowledge to use these services.

[Cognitive Services](#) is a group of services, each supporting different, generalized prediction capabilities. The services are divided into different categories to help you find the right service.

**COGNITIVE SERVICES AND MACHINE LEARNING**

Service category	Purpose
<a href="#">Decision</a>	Build apps that surface recommendations for informed and efficient decision-making.
<a href="#">Language</a>	Allow your apps to process natural language with pre-built scripts, evaluate sentiment and learn how to recognize what users want.
<a href="#">Search</a>	Add Bing Search APIs to your apps and harness the ability to comb billions of webpages, images, videos, and news with a single API call.
<a href="#">Speech</a>	Convert speech into text and text into natural-sounding speech. Translate from one language to another and enable speaker verification and recognition.
<a href="#">Vision</a>	Recognize, identify, caption, index, and moderate your pictures, videos, and digital ink content.

Use Cognitive Services when you:

- Can use a generalized solution.
- Access solution from a programming REST API or SDK.

Use another machine-learning solution when you:

- Need to choose the algorithm and need to train on very specific data.

## **What is machine learning?**

Machine learning is a concept where you bring together data and an algorithm to solve a specific need. Once the data and algorithm are trained, the output is a model that you can use again with different data. The trained model provides insights based on the new data.

The process of building a machine learning system requires some knowledge of machine learning or data science.

Machine learning is provided using [Azure Machine Learning \(AML\) products and services](#).

## **What is a Cognitive Service?**

A Cognitive Service provides part or all of the components in a machine learning solution: data, algorithm, and trained model. These services are meant to require general knowledge about your data without needing experience with machine learning or data science. These services provide both REST API(s) and language-based SDKs. As a result, you need to have programming language knowledge to use the services.

## **How are Cognitive Services and Azure Machine Learning (AML) similar?**

Both have the end-goal of applying artificial intelligence (AI) to enhance business operations, though how each provides this in the respective offerings is different.

Generally, the audiences are different:

- Cognitive Services are for developers without machine-learning experience.
- Azure Machine Learning is tailored for data scientists.

## **How is a Cognitive Service different from machine learning?**

A Cognitive Service provides a trained model for you. This brings data and an algorithm together, available from a REST API(s) or SDK. You can implement this service within minutes, depending on your scenario. A Cognitive Service provides answers to general problems such as key phrases in text or item identification in images.

Machine learning is a process that generally requires a longer period of time to implement successfully. This time is spent on data collection, cleaning, transformation, algorithm selection, model training, and deployment to get to the same level of functionality provided by a Cognitive Service. With machine learning, it is possible to provide answers to highly specialized and/or specific problems. Machine learning problems require familiarity with the specific subject matter and data of the problem under consideration, as well as expertise in data science.

## What kind of data do you have?

Cognitive Services, as a group of services, can require none, some, or all custom data for the trained model.

### No additional training data required

Services that provide a fully-trained model can be treated as a *opaque box*. You don't need to know how they work or what data was used to train them. You bring your data to a fully trained model to get a prediction.

### Some or all training data required

Some services allow you to bring your own data, then train a model. This allows you to extend the model using the Service's data and algorithm with your own data. The output matches your needs. When you bring your own data, you may need to tag the data in a way specific to the service. For example, if you are training a model to identify flowers, you can provide a catalog of flower images along with the location of the flower in each image to train the model.

A service may *allow* you to provide data to enhance its own data. A service may *require* you to provide data.

### Real-time or near real-time data required

A service may need real-time or near-real time data to build an effective model. These services process significant amounts of model data.

## Service requirements for the data model

The following data categorizes each service by which kind of data it allows or requires.

SERVICE REQUIREMENTS FOR THE DATA MODEL			
Cognitive Service	No training data required	You provide some or all training data	Real-time or near real-time data collection
<a href="#">Anomaly Detector</a>	x	x	x
Bing Search	x		
<a href="#">Computer Vision</a>	x		
<a href="#">Content Moderator</a>	x		x
<a href="#">Custom Vision</a>		x	
<a href="#">Face</a>	x	x	
<a href="#">Ink Recognizer</a>	x	x	

### SERVICE REQUIREMENTS FOR THE DATA MODEL

Cognitive Service	No training data required	You provide some or all training data	Real-time or near real-time data collection
<a href="#">Language Understanding (LUIS)</a>		x	
<a href="#">Personalizer</a>	x*	x*	x
<a href="#">QnA Maker</a>		x	
<a href="#">Speaker Recognizer</a>		x	
<a href="#">Speech Text-to-speech (TTS)</a>	x	x	
<a href="#">Speech Speech-to-text (STT)</a>	x	x	
<a href="#">Speech Translation</a>	x		
<a href="#">Language service</a>	x		
<a href="#">Translator</a>	x		
<a href="#">Translator - custom translator</a>		x	

\*Personalizer only needs training data collected by the service (as it operates in real-time) to evaluate your policy and data. Personalizer does not need large historical datasets for up-front or batch training.

### **Where can you use Cognitive Services?**

The services are used in any application that can make REST API(s) or SDK calls. Examples of applications include web sites, bots, virtual or mixed reality, desktop and mobile applications.

### **How is Azure Cognitive Search related to Cognitive Services?**

[Azure Cognitive Search](#) is a separate cloud search service that optionally uses Cognitive Services to add image and natural language processing to indexing workloads. Cognitive Services is exposed in Azure Cognitive Search through [built-in skills](#) that wrap individual APIs. You can use a free resource for walkthroughs, but plan on creating and attaching a [billable resource](#) for larger volumes.

### **How can you use Cognitive Services?**

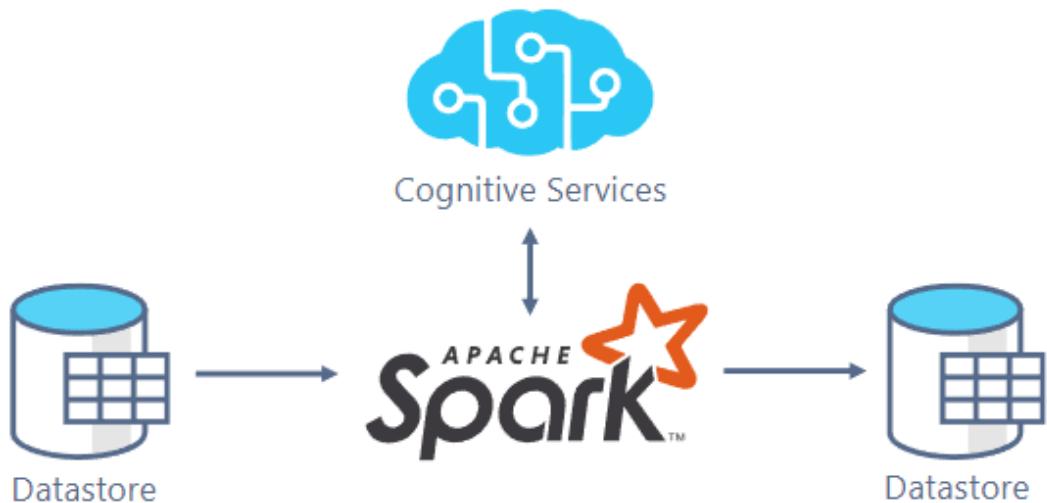
Each service provides information about your data. You can combine services together to chain solutions such as converting speech (audio) to text, translating the text into many

languages, then using the translated languages to get answers from a knowledge base. While Cognitive Services can be used to create intelligent solutions on their own, they can also be combined with traditional machine learning projects to supplement models or accelerate the development process.

Cognitive Services that provide exported models for other machine learning tools:

HOW CAN YOU USE COGNITIVE SERVICES?	
Cognitive	Model information
<a href="#">Custom Vision</a>	<a href="#">Export</a> for Tensorflow for Android, CoreML for iOS11, ONNX for Windows ML

## Azure Cognitive Services for Big Data



The Azure Cognitive Services for Big Data lets users channel terabytes of data through Cognitive Services using [Apache Spark™](#). With the Cognitive Services for Big Data, it's easy to create large-scale intelligent applications with any datastore.

With Cognitive Services for Big Data you can embed continuously improving, intelligent models directly into Apache Spark™ and SQL computations. These tools liberate developers from low-level networking details, so that they can focus on creating smart, distributed applications.

## **Features and benefits**

Cognitive Services for Big Data can use services from any region in the world, as well as [containerized Cognitive Services](#). Containers support low or no connectivity deployments with ultra-low latency responses. Containerized Cognitive Services can be run locally, directly on the worker nodes of your Spark cluster, or on an external orchestrator like Kubernetes.

## **Supported services**

[Cognitive Services](#), accessed through APIs and SDKs, help developers build intelligent applications without having AI or data science skills. With Cognitive Services you can make your applications see, hear, speak, understand, and reason. To use the Cognitive Services, your application must send data to the service over the network. Once received, the service sends an intelligent response in return. The following services are available for big data workloads:

## Vision

VISION	
Service Name	Service Description
<a href="#">Computer Vision</a>	The Computer Vision service provides you with access to advanced algorithms for processing images and returning information.
<a href="#">Face</a>	The Face service provides access to advanced face algorithms, enabling face attribute detection and recognition.

## Speech

SPEECH	
Service Name	Service Description
<a href="#">Speech service</a>	The Speech service provides access to features like speech recognition, speech synthesis, speech translation, and speaker verification and identification.

## Decision

DECISION	
Service	Service Description
<a href="#">Anomaly Detector</a>	The Anomaly Detector (Preview) service allows you to monitor and detect abnormalities in your time series data.

## Language

LANGUAGE	
Service Name	Service Description
<a href="#">Language service</a>	The Language service provides natural language processing over raw text for sentiment analysis, key-phrase extraction, and language detection.

## Search

SEARCH	
Service Name	Service Description
<a href="#">Bing Image Search</a>	The Bing Image Search service returns a display of images determined to be relevant to the user's query.

## Supported programming languages for Cognitive Services for Big Data

The Cognitive Services for Big Data are built on Apache Spark. Apache Spark is a distributed computing library that supports Java, Scala, Python, R, and many other languages. These languages are currently supported.

### Python

We provide a PySpark API in the `mmlspark.cognitive` namespace of [Microsoft ML for Apache Spark](#).

### Scala and Java

We provide a Scala and Java-based Spark API in the `com.microsoft.ml.spark.cognitive` namespace of [Microsoft ML for Apache Spark](#). For more information, see the [Scala Developer API](#).

## Supported platforms and connectors

The Cognitive Services for Big Data requires Apache Spark. There are several Apache Spark platforms that support the Cognitive Services for Big Data.

### Azure Databricks

[Azure Databricks](#) is an Apache Spark-based analytics platform optimized for the Microsoft Azure cloud services platform. It provides one-click setup, streamlined work-flows, and an interactive workspace that supports collaboration between data scientists, data engineers, and business analysts.

### Azure Synapse Analytics

[Azure Synapse Analytics](#) is an enterprise data warehouse that uses massive parallel processing. With Synapse Analytics, you can quickly run complex queries across petabytes of data. Azure Synapse Analytics provides managed Spark Pools to run Spark Jobs with an intuitive Jupyter Notebook Interface.

### Azure Kubernetes Service

[Azure Kubernetes Service \(AKS\)](#) orchestrates Docker Containers and distributed applications at massive scales. AKS is a managed Kubernetes offering that simplifies using Kubernetes in Azure. Kubernetes can enable fine-grained control of Cognitive Service scale, latency, and networking. However, we recommend using Azure Databricks or Azure Synapse Analytics if you're unfamiliar with Apache Spark.

## Data Connectors

Once you have a Spark Cluster, the next step is connecting to your data. Apache Spark has a broad collection of database connectors. These connectors allow applications to work with large datasets no matter where they're stored.

## Concepts

### Spark

[Apache Spark™](#) is a unified analytics engine for large-scale data processing. Its parallel processing framework boosts performance of big data and analytic applications. Spark can operate as both a batch and stream processing system, without changing core application code.

The basis of Spark is the DataFrame: a tabular collection of data distributed across the Apache Spark worker nodes. A Spark DataFrame is like a table in a relational database or a data frame in R/Python, but with limitless scale. DataFrames can be constructed from many sources such as: structured data files, tables in Hive, or external databases. Once your data is in a Spark DataFrame, you can:

- Do SQL-style computations such as join and filter tables.
- Apply functions to large datasets using MapReduce style parallelism.
- Apply Distributed Machine Learning using Microsoft Machine Learning for Apache Spark.
- Use the Cognitive Services for Big Data to enrich your data with ready-to-use intelligent services.

### Microsoft Machine Learning for Apache Spark (MMLSpark)

[Microsoft Machine Learning for Apache Spark](#) (MMLSpark) is an open-source, distributed machine learning library (ML) built on Apache Spark. The Cognitive Services for Big Data is included in this package. Additionally, MMLSpark contains several other ML tools for Apache Spark, such as LightGBM, Vowpal Wabbit, OpenCV, LIME, and more. With MMLSpark, you can build powerful predictive and analytical models from any Spark datasource.

### HTTP on Spark

Cognitive Services for Big Data is an example of how we can integrate intelligent web services with big data. Web services power many applications across the globe and most services communicate through the Hypertext Transfer Protocol (HTTP). To work with *arbitrary* web services at large scales, we provide HTTP on Spark. With HTTP on Spark, you can pass terabytes of data through any web service. Under the hood, we use this technology to power Cognitive Services for Big Data.

# AZURE MACHINE LEARNING

## What is Azure Machine Learning?

Azure Machine Learning is a cloud service for accelerating and managing the machine learning project lifecycle. Machine learning professionals, data scientists, and engineers can use it in their day-to-day workflows: Train and deploy models, and manage MLOps.

You can create a model in Azure Machine Learning or use a model built from an open-source platform, such as Pytorch, TensorFlow, or scikit-learn. MLOps tools help you monitor, retrain, and redeploy models.

## Who is Azure Machine Learning for?

Azure Machine Learning is for individuals and teams implementing MLOps within their organization to bring machine learning models into production in a secure and auditable production environment.

Data scientists and ML engineers will find tools to accelerate and automate their day-to-day workflows. Application developers will find tools for integrating models into applications or services. Platform developers will find a robust set of tools, backed by durable Azure Resource Manager APIs, for building advanced ML tooling.

Enterprises working in the Microsoft Azure cloud will find familiar security and role-based access control (RBAC) for infrastructure. You can set up a project to deny access to protected data and select operations.

## Collaboration for machine learning teams

Machine learning projects often require a team with varied skillsets to build and maintain. Azure Machine Learning has tools that help enable collaboration, such as:

- Shared notebooks, compute resources, data, and environments
- Tracking and auditability that shows who made changes and when
- Asset versioning

## Tools for developers

Developers find familiar interfaces in Azure Machine Learning, such as:

- [Python SDK](#)
- [Azure Resource Manager REST APIs \(preview\)](#)
- [CLI v2 \(preview\)](#)

## Studio UI

The [Azure Machine Learning studio](#) is a graphical user interface for a project workspace. In the studio, you can:

- View runs, metrics, logs, outputs, and so on.
- Author and edit notebooks and files.
- Manage common assets, such as
  - Data credentials
  - Compute
  - Environments
- Visualize run metrics, results, and reports.
- Visualize pipelines authored through developer interfaces.
- Author AutoML jobs.

Plus, the designer has a drag-and-drop interface where you can train and deploy models.

## Enterprise-readiness and security

Azure Machine Learning integrates with the Azure cloud platform to add security to ML projects. Security integrations include:

- Azure Virtual Networks (VNets) with network security groups
- Azure Key Vault where you can save security secrets, such as access information for storage accounts
- Azure Container Registry set up behind a VNet

## Azure integrations for complete solutions

Other integrations with Azure services support a machine learning project from end-to-end. They include:

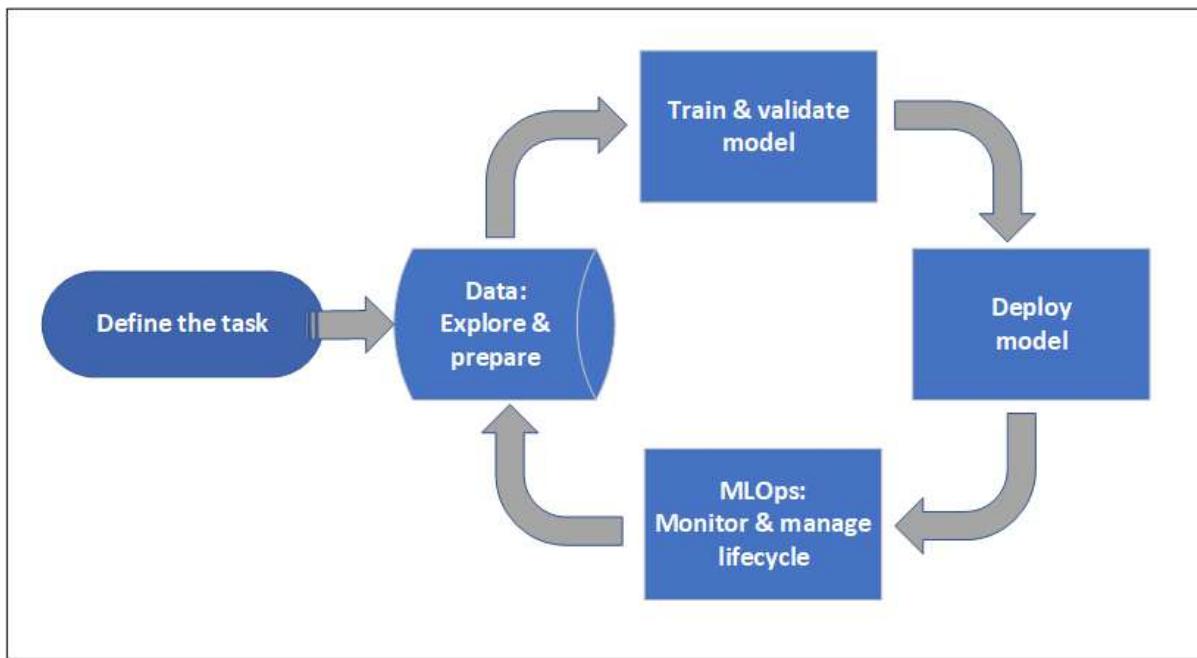
- Azure Synapse Analytics to process and stream data with Spark
- Azure Arc, where you can run Azure services in a Kubernetes environment
- Storage and database options, such as Azure SQL Database, Azure Storage Blobs, and so on
- Azure App Service allowing you to deploy and manage ML-powered apps

## Machine learning project workflow

Typically models are developed as part of a project with an objective and goals. Projects often involve more than one person. When experimenting with data, algorithms, and models, development is iterative.

## Project lifecycle

While the project lifecycle can vary by project, it will often look like this:



A workspace organizes a project and allows for collaboration for many users all working toward a common objective. Users in a workspace can easily share the results of their runs from experimentation in the studio user interface or use versioned assets for jobs like environments and storage references.

When a project is ready for operationalization, users' work can be automated in a machine learning pipeline and triggered on a schedule or HTTPS request.

Models can be deployed to the managed inferencing solution, for both real-time and batch deployments, abstracting away the infrastructure management typically required for deploying models.

## Train models

In Azure Machine Learning, you can run your training script in the cloud or build a model from scratch. Customers often bring models they've built and trained in open-source frameworks, so they can operationalize them in the cloud.

## Open and interoperable

Data scientists can use models in Azure Machine Learning that they've created in common Python frameworks, such as:

- PyTorch
- TensorFlow
- scikit-learn
- XGBoost
- LightGBM

Other languages and frameworks are supported as well, including:

- R
- .NET

## Automated featurization and algorithm selection (AutoML)

In a repetitive, time-consuming process, in classical machine learning data scientists use prior experience and intuition to select the right data featurization and algorithm for training. Automated ML (AutoML) speeds this process and can be used through the studio UI or Python SDK.

## Hyperparameter optimization

Hyperparameter optimization, or hyperparameter tuning, can be a tedious task. Azure Machine Learning can automate this task for arbitrary parameterized commands with little modification to your job definition. Results are visualized in the studio.

## Multinode distributed training

Efficiency of training for deep learning and sometimes classical machine learning training jobs can be drastically improved via multinode distributed training. Azure Machine Learning compute clusters offer the latest GPU options.

Supported via Azure Arc-attached Kubernetes (preview) and Azure ML compute clusters:

- PyTorch
- TensorFlow
- MPI

The MPI distribution can be used for Horovod or custom multinode logic. Additionally, Apache Spark is supported via Azure Synapse Analytics Spark clusters (preview).

## Embarrassingly parallel training

Scaling a machine learning project may require scaling embarrassingly parallel model training. This pattern is common for scenarios like forecasting demand, where a model may be trained for many stores.

## Deploy models

To bring a model into production, it is deployed. Azure Machine Learning's managed endpoints abstract the required infrastructure for both batch or real-time (online) model scoring (inferencing).

## Real-time and batch scoring (inferencing)

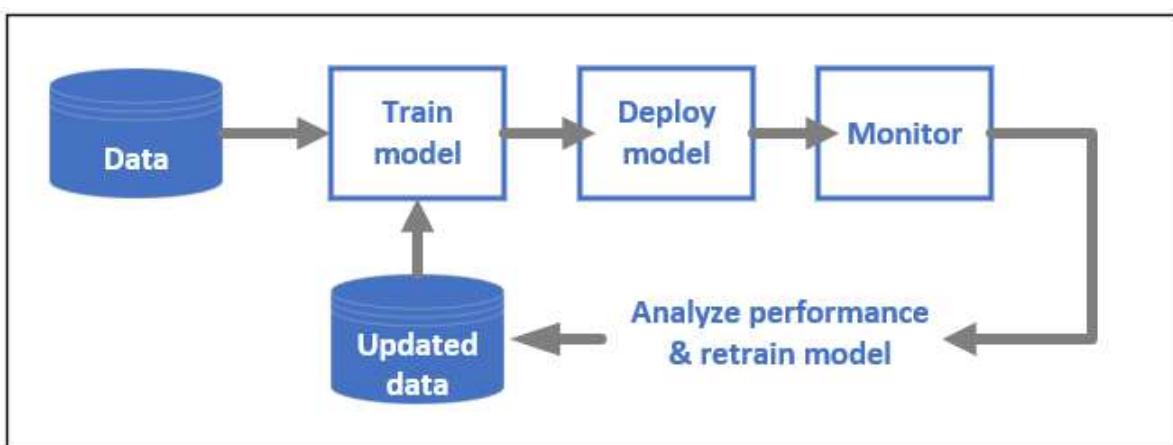
*Batch scoring*, or *batch inferencing*, involves invoking an endpoint with a reference to data. The batch endpoint runs jobs asynchronously to process data in parallel on compute clusters and store the data for further analysis.

*Real-time scoring*, or *online inferencing*, involves invoking an endpoint with one or more model deployments and receiving a response in near-real-time via HTTPs. Traffic can be split across multiple deployments, allowing for testing new model versions by diverting some amount of traffic initially and increasing once confidence in the new model is established.

## MLOps: DevOps for machine learning

DevOps for machine learning models, often called MLOps, is a process for developing models for production. A model's lifecycle from training to deployment must be auditable if not reproducible.

### ML model lifecycle



### Integrations enabling MLOPs

Azure Machine Learning is built with the model lifecycle in mind. You can audit the model lifecycle down to a specific commit and environment.

Some key features enabling MLOps include:

- git integration
- MLflow integration
- Machine learning pipeline scheduling
- Azure Event Grid integration for custom triggers
- Easy to use with CI/CD tools like GitHub Actions or Azure DevOps

Also, Azure Machine Learning includes features for monitoring and auditing:

- Job artifacts, such as code snapshots, logs, and other outputs
- Lineage between jobs and assets, such as containers, data, and compute resources

# What is Azure Machine Learning studio?

In this article, you learn about Azure Machine Learning studio, the web portal for data scientist developers in [Azure Machine Learning](#). The studio combines no-code and code-first experiences for an inclusive data science platform.

## Author machine learning projects

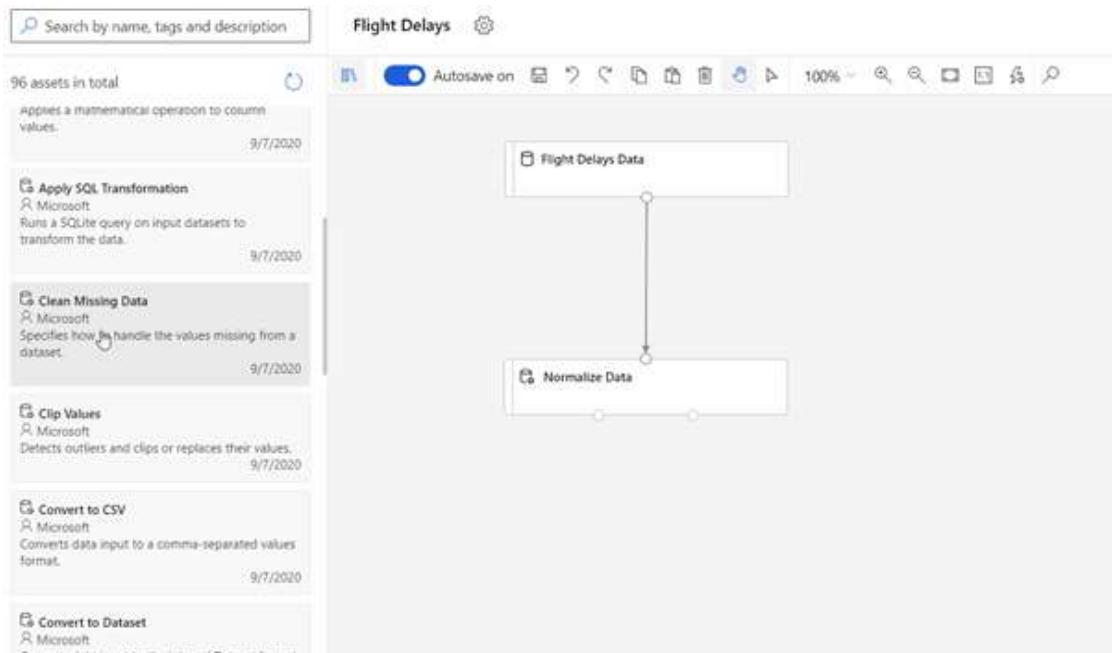
The studio offers multiple authoring experiences depending on the type project and the level of user experience.

- **Notebooks**

Write and run your own code in managed [Jupyter Notebook servers](#) that are directly integrated in the studio.

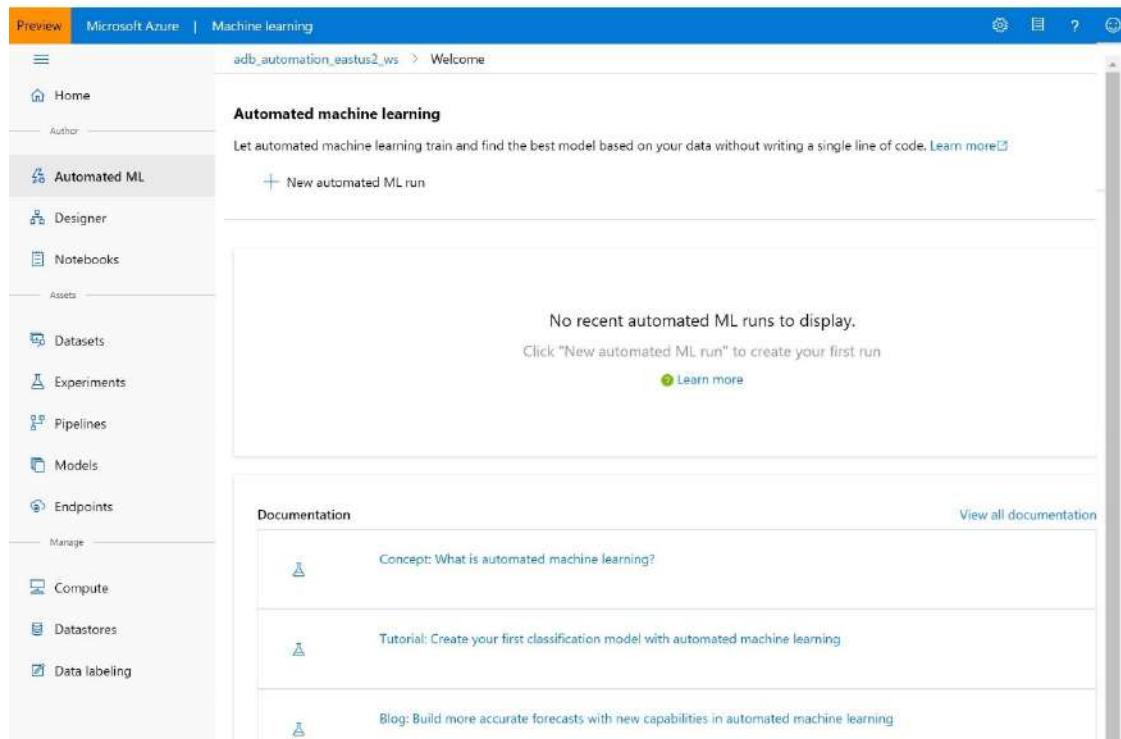
- **Azure Machine Learning designer**

Use the designer to train and deploy machine learning models without writing any code. Drag and drop datasets and components to create ML pipelines.



- **Automated machine learning UI**

Learn how to create [automated ML experiments](#) with an easy-to-use interface.



- **Data labeling**

Use Azure Machine Learning data labeling to efficiently coordinate [image labeling](#) or [text labeling](#) projects.

## Manage assets and resources

Manage your machine learning assets directly in your browser. Assets are shared in the same workspace between the SDK and the studio for a seamless experience. Use the studio to manage:

- Models
- Datasets
- Datastores
- Compute resources
- Notebooks
- Experiments
- Run logs
- Pipelines
- Pipeline endpoints

Even if you're an experienced developer, the studio can simplify how you manage workspace resources.

## ML Studio (classic) vs Azure Machine Learning studio

Released in 2015, **ML Studio (classic)** was the first drag-and-drop machine learning model builder in Azure. **ML Studio (classic)** is a standalone service that only offers a visual experience. Studio (classic) does not interoperate with Azure Machine Learning.

**Azure Machine Learning** is a separate, and modernized, service that delivers a complete data science platform. It supports both code-first and low-code experiences.

**Azure Machine Learning studio** is a web portal *in* Azure Machine Learning that contains low-code and no-code options for project authoring and asset management. If you're a new user, choose **Azure Machine Learning**, instead of ML Studio (classic). As a complete ML platform, Azure Machine Learning offers:

- Scalable compute clusters for large-scale training.
- Enterprise security and governance.
- Interoperable with popular open-source tools.
- End-to-end MLOps.

### Feature comparison

The following table summarizes the key differences between ML Studio (classic) and Azure Machine Learning.

FEATURE COMPARISON		
Feature	ML Studio (classic)	Azure Machine Learning
Drag and drop interface	Classic experience	Updated experience - <a href="#">Azure Machine Learning designer</a>
Code SDKs	Not supported	Fully integrated with <a href="#">Azure Machine Learning Python</a> and <a href="#">R</a> SDKs
Experiment	Scalable (10-GB training data limit)	Scale with compute target
Training compute targets	Proprietary compute target, CPU support only	Wide range of customizable <a href="#">training compute targets</a> . Includes GPU and CPU support
Deployment compute targets	Proprietary web service format, not customizable	Wide range of customizable <a href="#">deployment compute targets</a> . Includes GPU and CPU support
ML Pipeline	Not supported	Build flexible, modular <a href="#">pipelines</a> to automate workflows
MLOps	Basic model management and deployment; CPU only deployments	Entity versioning (model, data, workflows), workflow automation, integration with CI/CD tooling, CPU and GPU deployments <a href="#">and more</a>
Model format	Proprietary format, Studio (classic) only	Multiple supported formats depending on training job type

## FEATURE COMPARISON

Feature	ML Studio (classic)	Azure Machine Learning
Automated model training and hyperparameter tuning	Not supported	<a href="#">Supported</a> . Code-first and no-code options.
Data drift detection	Not supported	<a href="#">Supported</a>
Data labeling projects	Not supported	<a href="#">Supported</a>
Role-Based Access Control (RBAC)	Only contributor and owner role	<a href="#">Flexible role definition and RBAC control</a>
AI Gallery	Supported ( <a href="https://gallery.azure.ai/">https://gallery.azure.ai/</a> )	Unsupported Learn with <a href="#">sample Python SDK notebooks</a> .

## Troubleshooting

- **Missing user interface items in studio** Azure role-based access control can be used to restrict actions that you can perform with Azure Machine Learning. These restrictions can prevent user interface items from appearing in the Azure Machine Learning studio. For example, if you are assigned a role that cannot create a compute instance, the option to create a compute instance will not appear in the studio.

## How Azure Machine Learning works: Architecture and concepts

### Workspace

A [machine learning workspace](#) is the top-level resource for Azure Machine Learning.



The workspace is the centralized place to:

- Manage resources you use for training and deployment of models, such as [computes](#)
- Store assets you create when you use Azure Machine Learning, including:
  - [Environments](#)
  - [Experiments](#)
  - [Pipelines](#)
  - [Datasets](#)
  - [Models](#)
  - [Endpoints](#)

A workspace includes other Azure resources that are used by the workspace:

- [Azure Container Registry \(ACR\)](#): Registers docker containers that you use during training and when you deploy a model. To minimize costs, ACR is only created when deployment images are created.
- [Azure Storage account](#): Is used as the default datastore for the workspace. Jupyter notebooks that are used with your Azure Machine Learning compute instances are stored here as well.
- [Azure Application Insights](#): Stores monitoring information about your models.
- [Azure Key Vault](#): Stores secrets that are used by compute targets and other sensitive information that's needed by the workspace.

You can share a workspace with others.

## Computes

A [compute target](#) is any machine or set of machines you use to run your training script or host your service deployment. You can use your local machine or a remote compute resource as a compute target. With compute targets, you can start training on your local machine and then scale out to the cloud without changing your training script.

Azure Machine Learning introduces two fully managed cloud-based virtual machines (VM) that are configured for machine learning tasks:

- **Compute instance:** A compute instance is a VM that includes multiple tools and environments installed for machine learning. The primary use of a compute instance is for your development workstation. You can start running sample notebooks with no setup required. A compute instance can also be used as a compute target for training and inferencing jobs.
- **Compute clusters:** Compute clusters are a cluster of VMs with multi-node scaling capabilities. Compute clusters are better suited for compute targets for large jobs and production. The cluster scales up automatically when a job is submitted. Use as a training compute target or for dev/test deployment.

## Datasets and datastores

Azure Machine Learning Datasets make it easier to access and work with your data. By creating a dataset, you create a reference to the data source location along with a copy of its metadata. Because the data remains in its existing location, you incur no extra storage cost, and don't risk the integrity of your data sources.

Datasets use [datastores](#) to securely connect to your Azure storage services. Datastores store connection information without putting your authentication credentials and the integrity of your original data source at risk. They store connection information, like your subscription ID and token authorization in your Key Vault associated with the workspace, so you can securely access your storage without having to hard code them in your script.

## Environments

### [Workspace](#) > Environments

An [environment](#) is the encapsulation of the environment where training or scoring of your machine learning model happens. The environment specifies the Python packages, environment variables, and software settings around your training and scoring scripts.

## Experiments

### [Workspace](#) > Experiments

An experiment is a grouping of many runs from a specified script. It always belongs to a workspace. When you submit a run, you provide an experiment name. Information for the run is stored under that experiment. If the name doesn't exist when you submit an experiment, a new experiment is automatically created.

## Runs

### [Workspace](#) > Experiments > Run

A run is a single execution of a training script. An experiment will typically contain multiple runs.

Azure Machine Learning records all runs and stores the following information in the experiment:

- Metadata about the run (timestamp, duration, and so on)
- Metrics that are logged by your script
- Output files that are autocollected by the experiment or explicitly uploaded by you
- A snapshot of the directory that contains your scripts, prior to the run

You produce a run when you submit a script to train a model. A run can have zero or more child runs. For example, the top-level run might have two child runs, each of which might have its own child run.

## Run configurations

[Workspace](#) > [Experiments](#) > [Run](#) > **Run configuration**

A run configuration defines how a script should be run in a specified compute target. You use the configuration to specify the script, the compute target and Azure ML environment to run on, any distributed job-specific configurations, and some additional properties. For more information on the full set of configurable options for runs, see [ScriptRunConfig](#).

A run configuration can be persisted into a file inside the directory that contains your training script. Or it can be constructed as an in-memory object and used to submit a run.

## Snapshots

[Workspace](#) > [Experiments](#) > [Run](#) > **Snapshot**

When you submit a run, Azure Machine Learning compresses the directory that contains the script as a zip file and sends it to the compute target. The zip file is then extracted, and the script is run there. Azure Machine Learning also stores the zip file as a snapshot as part of the run record. Anyone with access to the workspace can browse a run record and download the snapshot.

## Logging

Azure Machine Learning automatically logs standard run metrics for you. However, you can also [use the Python SDK to log arbitrary metrics](#).

There are multiple ways to view your logs: monitoring run status in real time, or viewing results after completion.

## Git tracking and integration

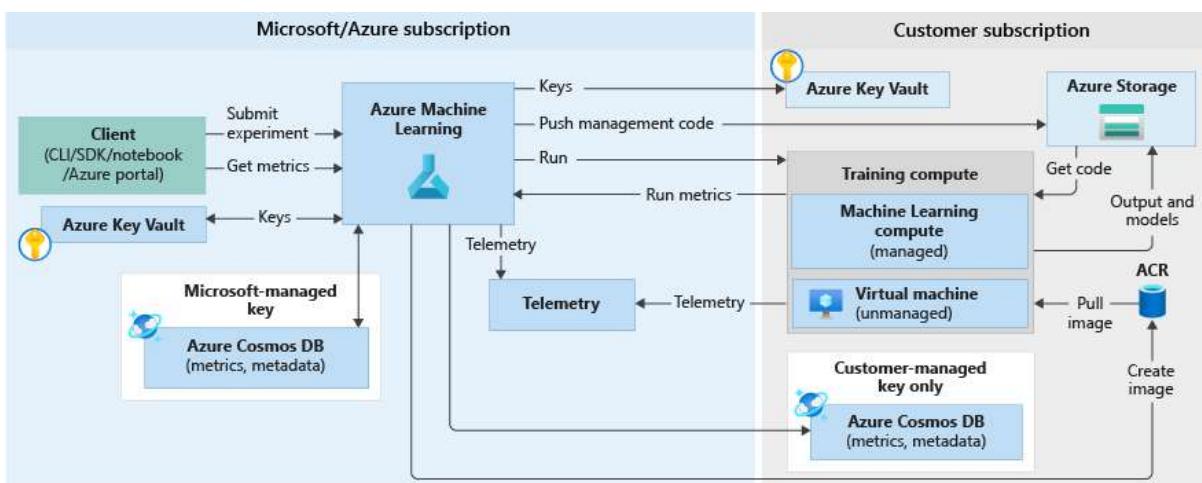
When you start a training run where the source directory is a local Git repository, information about the repository is stored in the run history. This works with runs submitted using a script run configuration or ML pipeline. It also works for runs submitted from the SDK or Machine Learning CLI.

## Training workflow

When you run an experiment to train a model, the following steps happen. These are illustrated in the training workflow diagram below:

- Azure Machine Learning is called with the snapshot ID for the code snapshot saved in the previous section.
- Azure Machine Learning creates a run ID (optional) and a Machine Learning service token, which is later used by compute targets like Machine Learning Compute/VMs to communicate with the Machine Learning service.

- You can choose either a managed compute target (like Machine Learning Compute) or an unmanaged compute target (like VMs) to run training jobs. Here are the data flows for both scenarios:
  - VMs/HDIInsight, accessed by SSH credentials in a key vault in the Microsoft subscription. Azure Machine Learning runs management code on the compute target that:
    - Prepares the environment. (Docker is an option for VMs and local computers. See the following steps for Machine Learning Compute to understand how running experiments on Docker containers works.)
    - Downloads the code.
    - Sets up environment variables and configurations.
    - Runs user scripts (the code snapshot mentioned in the previous section).
  - Machine Learning Compute, accessed through a workspace-managed identity. Because Machine Learning Compute is a managed compute target (that is, it's managed by Microsoft) it runs under your Microsoft subscription.
    - Remote Docker construction is kicked off, if needed.
    - Management code is written to the user's Azure Files share.
    - The container is started with an initial command. That is, management code as described in the previous step.
- After the run completes, you can query runs and metrics. In the flow diagram below, this step occurs when the training compute target writes the run metrics back to Azure Machine Learning from storage in the Cosmos DB database. Clients can call Azure Machine Learning. Machine Learning will in turn pull metrics from the Cosmos DB database and return them back to the client.



## Models

At its simplest, a model is a piece of code that takes an input and produces output. Creating a machine learning model involves selecting an algorithm, providing it with data, and [tuning hyperparameters](#). Training is an iterative process that produces a trained model, which encapsulates what the model learned during the training process.

You can bring a model that was trained outside of Azure Machine Learning. Or you can train a model by submitting a [run](#) of an [experiment](#) to a [compute target](#) in Azure Machine Learning. Once you have a model, you [register the model](#) in the workspace.

Azure Machine Learning is framework agnostic. When you create a model, you can use any popular machine learning framework, such as Scikit-learn, XGBoost, PyTorch, TensorFlow, and Chainer.

## Model registry

[Workspace](#) > [Models](#)

The **model registry** lets you keep track of all the models in your Azure Machine Learning workspace.

Models are identified by name and version. Each time you register a model with the same name as an existing one, the registry assumes that it's a new version. The version is incremented, and the new model is registered under the same name.

When you register the model, you can provide additional metadata tags and then use the tags when you search for models.

### Tip

A registered model is a logical container for one or more files that make up your model. For example, if you have a model that is stored in multiple files, you can register them as a single model in your Azure Machine Learning workspace. After registration, you can then download or deploy the registered model and receive all the files that were registered.

You can't delete a registered model that is being used by an active deployment.

## Deployment

You deploy a [registered model](#) as a service endpoint. You need the following components:

- **Environment.** This environment encapsulates the dependencies required to run your model for inference.
- **Scoring code.** This script accepts requests, scores the requests by using the model, and returns the results.
- **Inference configuration.** The inference configuration specifies the environment, entry script, and other components needed to run the model as a service.

For more information about these components, see [Deploy models with Azure Machine Learning](#).

## Endpoints

[Workspace](#) > [Endpoints](#)

An endpoint is an instantiation of your model into a web service that can be hosted in the cloud.

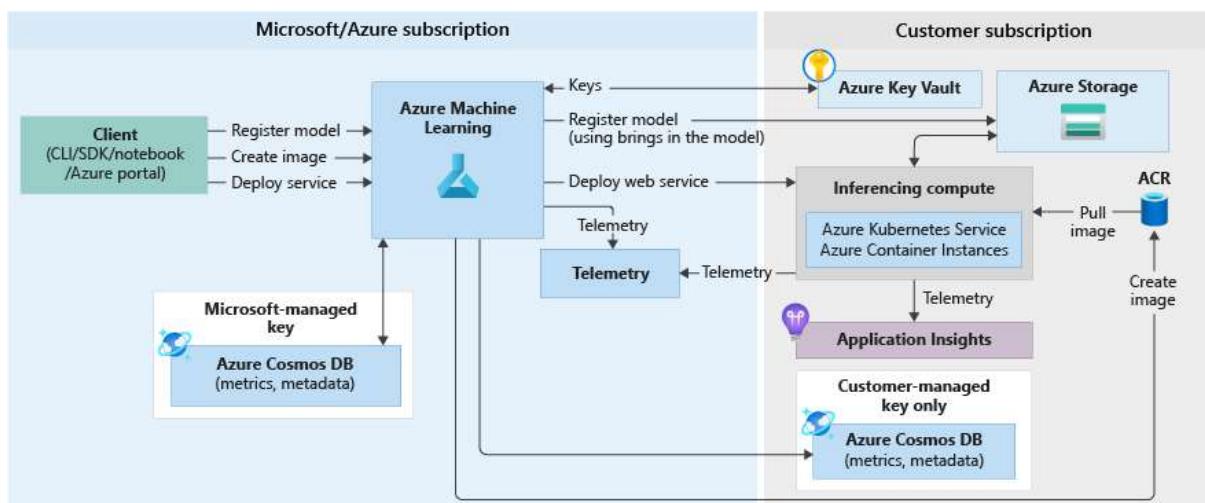
## Web service endpoint

When deploying a model as a web service, the endpoint can be deployed on Azure Container Instances, Azure Kubernetes Service, or FPGAs. You create the service from your model, script, and associated files. These are placed into a base container image, which contains the execution environment for the model. The image has a load-balanced, HTTP endpoint that receives scoring requests that are sent to the web service.

You can enable Application Insights telemetry or model telemetry to monitor your web service. The telemetry data is accessible only to you. It's stored in your Application Insights and storage account instances. If you've enabled automatic scaling, Azure automatically scales your deployment.

The following diagram shows the inference workflow for a model deployed as a web service endpoint: Here are the details:

- The user registers a model by using a client like the Azure Machine Learning SDK.
- The user creates an image by using a model, a score file, and other model dependencies.
- The Docker image is created and stored in Azure Container Registry.
- The web service is deployed to the compute target (Container Instances/AKS) using the image created in the previous step.
- Scoring request details are stored in Application Insights, which is in the user's subscription.
- Telemetry is also pushed to the Microsoft/Azure subscription.



## Real-time endpoints

When you deploy a trained model in the designer, you can [deploy the model as a real-time endpoint](#). A real-time endpoint commonly receives a single request via the REST endpoint and returns a prediction in real-time. This is in contrast to batch processing, which processes multiple values at once and saves the results after completion to a datastore.

## *Pipeline endpoints*

Pipeline endpoints let you call your [ML Pipelines](#) programmatically via a REST endpoint. Pipeline endpoints let you automate your pipeline workflows. A pipeline endpoint is a collection of published pipelines. This logical organization lets you manage and call multiple pipelines using the same endpoint. Each published pipeline in a pipeline endpoint is versioned. You can select a default pipeline for the endpoint, or specify a version in the REST call.

## Automation

### Azure Machine Learning CLI

The [Azure Machine Learning CLI](#) is an extension to the Azure CLI, a cross-platform command-line interface for the Azure platform. This extension provides commands to automate your machine learning activities.

### ML Pipelines

You use [machine learning pipelines](#) to create and manage workflows that stitch together machine learning phases. For example, a pipeline might include data preparation, model training, model deployment, and inference/scoring phases. Each phase can encompass multiple steps, each of which can run unattended in various compute targets.

Pipeline steps are reusable, and can be run without rerunning the previous steps if the output of those steps hasn't changed. For example, you can retrain a model without rerunning costly data preparation steps if the data hasn't changed. Pipelines also allow data scientists to collaborate while working on separate areas of a machine learning workflow.

### Monitoring and logging

Azure Machine Learning provides the following monitoring and logging capabilities:

- For **Data Scientists**, you can monitor your experiments and log information from your training runs. For more information, see the following articles:
  - [Start, monitor, and cancel training runs](#)
  - [Log metrics for training runs](#)
  - [Track experiments with MLflow](#)
  - [Visualize runs with TensorBoard](#)
- For **Administrators**, you can monitor information about the workspace, related Azure resources, and events such as resource creation and deletion by using Azure Monitor.
- For **DevOps** or **MLOps**, you can monitor information generated by models deployed as web services to identify problems with the deployments and gather data submitted to the service. Interacting with your workspace

## Studio

[Azure Machine Learning studio](#) provides a web view of all the artifacts in your workspace. You can view results and details of your datasets, experiments, pipelines, models, and endpoints. You can also manage compute resources and datastores in the studio.

The studio is also where you access the interactive tools that are part of Azure Machine Learning:

- [Azure Machine Learning designer](#) to perform workflow steps without writing code
- Web experience for [automated machine learning](#)
- [Azure Machine Learning notebooks](#) to write and run your own code in integrated Jupyter notebook servers.
- Data labeling projects to create, manage, and monitor projects for labeling [images](#) or [text](#).
- 

## Programming tools

Interact with the service in any Python environment with the [Azure Machine Learning SDK for Python](#).

- Use [Azure Machine Learning designer](#) to perform the workflow steps without writing code.
- Use [Azure Machine Learning CLI](#) for automation.

# Microsoft Genomics

## What is Microsoft Genomics?

Microsoft Genomics offers a cloud implementation of the Burrows-Wheeler Aligner (BWA) and the Genome Analysis Toolkit (GATK) for secondary analysis. The service is ISO-certified and compliant with HIPAA regulations, and offers price predictability for your genome sequencing needs. Learn how to use the Microsoft Genomics service and integrate with our API by reading our quickstarts, tutorials, and documentation.

## Support your most demanding sequencing needs

Instead of managing your own datacenters, take advantage of the scale and experience of Microsoft in running exabyte-scale workloads. Our cloud implementation of the BWA-GATK is highly concordant with the Broad Institute's best practices pipeline.

## Keep your business running

Microsoft Genomics offers a 99.99% availability service level agreement (SLA) for receiving workflow requests.

## Secure your data

The Microsoft Genomics service is ISO 27001, ISO 27018, and ISO 9001 certified and compliant with HIPAA regulations.

# Quickstart: Run a workflow through the Microsoft Genomics service

In this quickstart, you upload input data into an Azure Blob storage account, and run a workflow through the Microsoft Genomics service by using the Python Genomics client. Microsoft Genomics is a scalable, secure service for secondary analysis that can rapidly process a genome, starting from raw reads and producing aligned reads and variant calls.

## Prerequisites

- An Azure account with an active subscription. [Create an account for free](#).
- [Python 2.7.12+](#), with pip installed, and python in your system path. The Microsoft Genomics client isn't compatible with Python 3.

## Set up: Create a Microsoft Genomics account in the Azure portal

To create a Microsoft Genomics account, navigate to [Create a Genomics account](#) in the Azure portal. If you don't have an Azure subscription yet, create one before creating a Microsoft Genomics account.

The screenshot shows the 'Create a Genomics account' wizard in the Azure portal. The top navigation bar includes 'Home > New > Marketplace > Everything > Genomics > Create a Genomics account'. The main title is 'Create a Genomics account'. Below it, there are three tabs: 'Basics' (selected), 'Tags', and 'Review + Create'. A descriptive text block explains that the service provides a cloud hosted solution for variant calling genomic samples, mentioning BWA/GATK pipeline efficiency. A 'Learn more' link is present. The 'PROJECT DETAILS' section asks to select a subscription and resource group. The 'Subscription' dropdown is empty. The 'Resource group' dropdown also has no options, with a 'Create new' link below it. The 'INSTANCE DETAILS' section requires an 'Account name' (with a placeholder 'Enter the name') and a 'Location' (dropdown menu). At the bottom, there are 'Review + Create' and 'Next : Tags' buttons.

Configure your Genomics account with the following information, as shown in the preceding image.

## **SET UP: CREATE A MICROSOFT GENOMICS ACCOUNT IN THE AZURE PORTAL**

Setting	Suggested value	Field description
Subscription	Your subscription name	This is the billing unit for your Azure services - For details about your subscription see <a href="#">Subscriptions</a>
Resource group	MyResourceGroup	Resource groups allow you to group multiple Azure resources (storage account, genomics account, etc.) into a single group for simple management. For more information, see <a href="#">Resource Groups</a> . For valid resource group names, see <a href="#">Naming Rules</a>
Account name	MyGenomicsAccount	Choose a unique account identifier. For valid names, see <a href="#">Naming Rules</a>
Location	West US 2	Service is available in West US 2, West Europe, and Southeast Asia

You can select **Notifications** in the top menu bar to monitor the deployment process.



## Set up: Install the Microsoft Genomics Python client

You need to install both Python and the Microsoft Genomics Python client `msgen` in your local environment.

### Install Python

The Microsoft Genomics Python client is compatible with Python 2.7.12 or a later 2.7.xx version. 2.7.14 is the suggested version.

### Important

Python 3.x isn't compatible with Python 2.7.xx. `msgen` is a Python 2.7 application. When running `msgen`, make sure that your active Python environment is using a 2.7.xx version of Python. You may get errors when trying to use `msgen` with a 3.x version of Python.

## Install the Microsoft Genomics Python client **msgen**

Use Python pip to install the Microsoft Genomics client `msgen`. The following instructions assume Python2.x is already in your system path. If you have issues with pip install not being recognized, you need to add Python and the scripts subfolder to your system path.

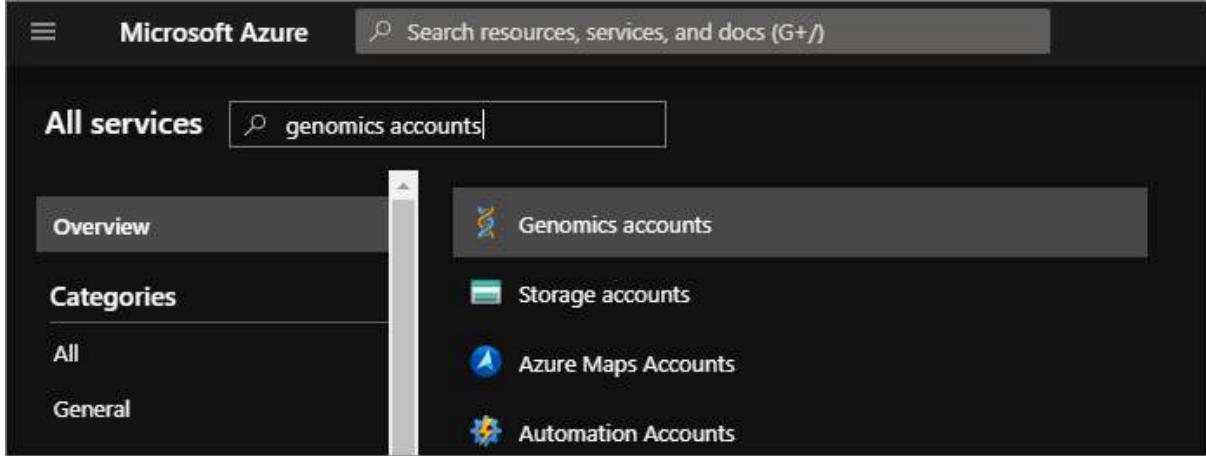
### Copy

```
pip install --upgrade --no-deps msgen  
pip install msgen
```

If you don't want to install `msgen` as a system-wide binary and modify system-wide Python packages, use the `--user` flag with pip. When you use the package-based installation or `setup.py`, all necessary required packages are installed.

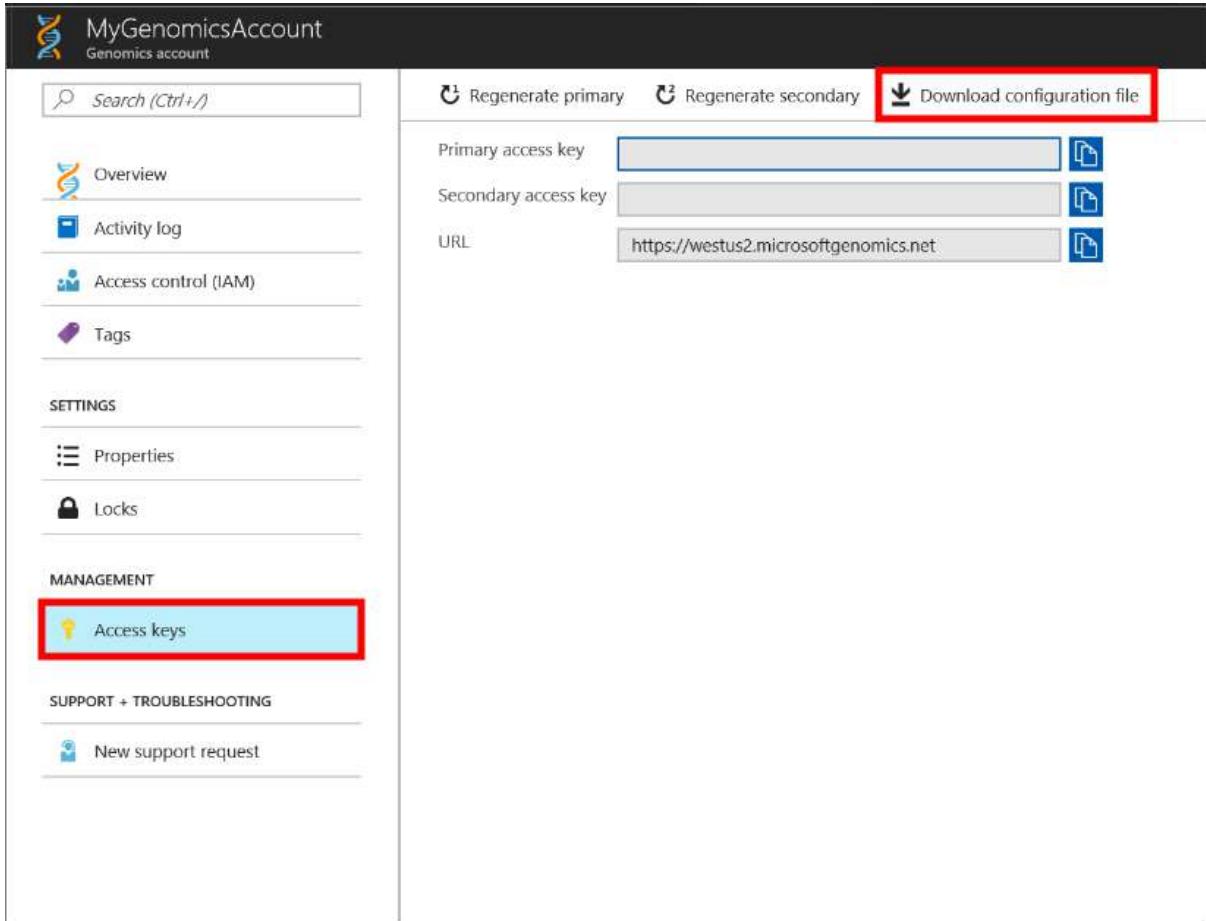
## Test msgen Python client

To test the Microsoft Genomics client, download the config file from your Genomics account. In the Azure portal, navigate to your Genomics account by selecting **All services** in the top left, and then searching for and selecting Genomics accounts.



The screenshot shows the Microsoft Azure portal's search interface. The search bar at the top contains the query "genomics accounts". Below the search bar, the "All services" section is visible, with a sidebar on the left showing categories like Overview, Categories, All, and General. The main pane displays a list of services: Genomics accounts (selected), Storage accounts, Azure Maps Accounts, and Automation Accounts. Each service item has a small icon to its left.

Select the Genomics account you just made, navigate to **Access Keys**, and download the configuration file.



The screenshot shows the "MyGenomicsAccount" settings page in the Azure portal. The left sidebar includes links for Overview, Activity log, Access control (IAM), Tags, Properties, Locks, and Access keys (which is highlighted with a red box). The main pane shows access key details: Primary access key, Secondary access key, and URL (https://westus2.microsoftgenomics.net). A "Download configuration file" button is located in the top right of this section, also highlighted with a red box.

Test that the Microsoft Genomics Python client is working with the following command

PythonCopy

```
msgen list -f "<full path where you saved the config file>"
```

## Create a Microsoft Azure Storage account

The Microsoft Genomics service expects inputs to be stored as block blobs in an Azure storage account. It also writes output files as block blobs to a user-specified container in an Azure storage account. The inputs and outputs can reside in different storage accounts. If you already have your data in an Azure storage account, you only need to make sure that it is in the same location as your Genomics account. Otherwise, egress charges are incurred when running the Microsoft Genomics service. If you don't yet have an Azure storage account, you need to create one and upload your data. You can find more information about Azure storage accounts [here](#), including what a storage account is and what services it provides. To create an Azure storage account, navigate to [Create storage account](#) in the Azure portal.

**Create storage account**

[Basics](#) [Advanced](#) [Tags](#) [Review + create](#)

Azure Storage is a Microsoft-managed service providing cloud storage that is highly available, secure, durable, scalable, and redundant. Azure Storage includes Azure Blobs (objects), Azure Data Lake Storage Gen2, Azure Files, Azure Queues, and Azure Tables. The cost of your storage account depends on the usage and the options you choose below. [Learn more](#)

**PROJECT DETAILS**

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

\* Subscription

\* Resource group  [Select existing...](#) [Create new](#)

**INSTANCE DETAILS**

The default deployment model is Resource Manager, which supports the latest Azure features. You may choose to deploy using the classic deployment model instead. [Choose classic deployment model](#)

\* Storage account name

\* Location

Performance  Standard  Premium

Account kind

Replication  [Read-access geo-redundant storage \(RA-GRS\)](#)

Access tier (default)   Cool  Hot

[Review + create](#) [Previous](#) [Next : Advanced >](#)

Configure your storage account with the following information, as shown in the preceding image. Use most of the standard options for a storage account, specifying only that the account is BlobStorage, not general purpose. Blob storage can be 2-5x faster for downloads and uploads. The default deployment model, Azure Resource Manager, is recommended.

## CREATE A MICROSOFT AZURE STORAGE ACCOUNT

Setting	Suggested value	Field description
Subscription	Your Azure subscription	For details about your subscription see <a href="#">Subscriptions</a>
Resource group	MyResourceGroup	You can select the same resource group as your Genomics account. For valid resource group names, see <a href="#">Naming rules</a>
Storage account name	MyStorageAccount	Choose a unique account identifier. For valid names, see <a href="#">Naming rules</a>
Location	West US 2	Use the same location as the location of your Genomics account, to reduce egress charges, and reduce latency.
Performance	Standard	The default is standard. For more details on standard and premium storage accounts, see <a href="#">Introduction to Microsoft Azure storage</a>
Account kind	BlobStorage	Blob storage can be 2-5x faster than general purpose for downloads and uploads.
Replication	Locally redundant storage	Locally redundant storage replicates your data within the datacenter in the region you created your storage account. For more information, see <a href="#">Azure Storage replication</a>
Access tier	Hot	Hot access indicates objects in the storage account will be more frequently accessed.

Then select **Review + create** to create your storage account. As you did with the creation of your Genomics account, you can select **Notifications** in the top menu bar to monitor the deployment process.

### Upload input data to your storage account

The Microsoft Genomics service expects paired end reads (fastq or bam files) as input files. You can choose to either upload your own data, or explore using publicly available sample data provided for you. If you would like to use the publicly available sample data, it is hosted here:

[https://msgensampled.blob.core.windows.net/small/chr21\\_1.fq.gz](https://msgensampled.blob.core.windows.net/small/chr21_1.fq.gz) [https://msgensampled.blob.core.windows.net/small/chr21\\_2.fq.gz](https://msgensampled.blob.core.windows.net/small/chr21_2.fq.gz)

Within your storage account, you need to make one blob container for your input data and a second blob container for your output data. Upload the input data into your input blob container. Various tools can be used to do this, including [Microsoft Azure Storage Explorer](#), [BlobPorter](#), or [AzCopy](#).

### Run a workflow through the Microsoft Genomics service using the **msgen** Python client

To run a workflow through the Microsoft Genomics service, edit the *config.txt* file to specify the input and output storage container for your data. Open the *config.txt* file that you downloaded

from your Genomics account. The sections you need to specify are your subscription key and the six items at the bottom, the storage account name, key, and container name for both the input and output. You can find this information by navigating in the Azure portal to **Access keys** for your storage account, or directly from the Azure Storage Explorer.

```
# Microsoft Genomics service - Command Line Interface - Configuration File
#
# Documentation: https://docs.microsoft.com/azure/genomics/
#
# Instructions
# 1. Entries are provided in key - value pairs, like key: value
# 2. Whitespace (tabs, spaces) don't matter
# 3. Lines starting with # are ignored
#
# Example usage:
#
# pip install msgen
# msgen submit -f c:\temp\config.txt -b1 sample_1.fq.gz -b2 sample_2.fq.gz
# msgen submit -f c:\temp\config.txt -b1 sample.bam

api_url_base: <Your Genomics Service API url here>
access_key: <Your Genomics account key here>

# Other available references (replace hg19m1 below): b37m1, hg19m1, hg38m1, hg38m1x
process_args: R=hg19m1

# Uncomment the appropriate process_name
process_name: snapgatk
#process_name: gatk4

poll: false

# To learn more about the optional "emit_ref_confidence" argument, see https://github.com/microsoft/msgen#release-notes-v080
# Uncomment the "emit_ref_confidence" argument below to produce "g.vcf" outputs.
#emit_ref_confidence: gvcf

# To learn more about the optional "bgzip_output" argument, see https://github.com/microsoft/msgen#release-notes-v090
# Uncomment the "bgzip_output" argument below to produce ".vcf.gz" and ".vcf.gz.tbi" outputs.
#bgzip_output: true

input_storage_account_name:
input_storage_account_key:
input_storage_account_container:
output_storage_account_name:
output_storage_account_key:
output_storage_account_container:
```

If you would like to run GATK4, set the process\_name parameter to gatk4.

By default, the Genomics service outputs VCF files. If you would like a gVCF output rather than a VCF output (equivalent to -emitRefConfidence in GATK 3.x and emit-ref-confidence in GATK 4.x), add the emit\_ref\_confidence parameter to your *config.txt* and set it to gvcf, as shown in the preceding figure. To change back to VCF output, either remove it from the *config.txt* file or set the emit\_ref\_confidence parameter to none.

bgzip is a tool that compresses the vcf or gvcf file, and tabix creates an index for the compressed file. By default, the Genomics service runs bgzip followed by tabix on ".g.vcf" output but does not run these tools by default for ".vcf" output. When run, the service produces ".gz" (bgzip output) and ".tbi" (tabix output) files. The argument is a boolean, which is set to false by default for ".vcf" output, and to true by default for ".g.vcf" output. To use on the command line, specify -bz or --bgzip-output as true (run bgzip and tabix) or false. To use this argument in the *config.txt* file, add bgzip\_output: true or bgzip\_output: false to the file.

Submit your workflow to the Microsoft Genomics service using the **msgen** Python client

Use the Microsoft Genomics Python client to submit your workflow with the following command:

Python

```
msgen submit -f [full path to your config file] -b1 [name of your first paired end read] -b2 [name of your second paired end read]
```

You can view the status of your workflows using the following command:

Python

```
msgen list -f c:\temp\config.txt
```

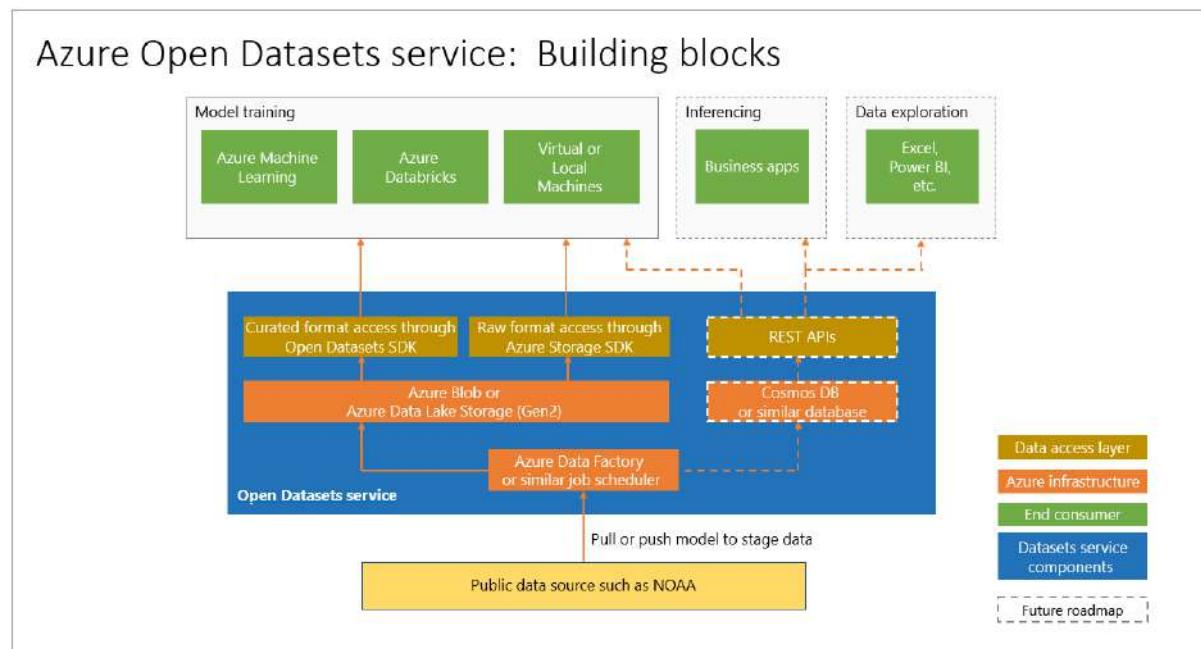
Once your workflow completes, you can view the output files in your Azure storage account in the output container that you configured.

## Azure Open Datasets

### What are Azure Open Datasets and how can you use them?

[Azure Open Datasets](#) are curated public datasets that you can use to add scenario-specific features to machine learning solutions for more accurate models. Open Datasets are in the cloud on Microsoft Azure and are integrated into Azure Machine Learning and readily available to Azure Databricks and Machine Learning Studio (classic). You can also access the datasets through APIs and use them in other products, such as Power BI and Azure Data Factory.

Datasets include public-domain data for weather, census, holidays, public safety, and location that help you train machine learning models and enrich predictive solutions. You can also share your public datasets on Azure Open Datasets.



#### Curated, prepared datasets

Curated open public datasets in Azure Open Datasets are optimized for consumption in machine learning workflows.

To see all the datasets available, go to the [Azure Open Datasets Catalog](#).

Data scientists often spend the majority of their time cleaning and preparing data for advanced analytics. Open Datasets are copied to the Azure cloud and preprocessed to save you time. At regular intervals data is pulled from the sources, such as by an FTP connection to the National Oceanic and Atmospheric Administration (NOAA). Next, data is parsed into a structured format, and then enriched as appropriate with features such as ZIP Code or location of the nearest weather station.

Datasets are cohosted with cloud compute in Azure making access and manipulation easier.

Following are examples of datasets available.

## Weather data

WEATHER DATA		
Dataset	Notebooks	Description
NOAA Integrated Surface Data (ISD)	Azure Notebooks Azure Databricks	Worldwide hourly weather data from NOAA with the best spatial coverage in North America, Europe, Australia, and parts of Asia. Updated daily.
NOAA Global Forecast System (GFS)	Azure Notebooks Azure Databricks	15-day U.S. hourly weather forecast data from NOAA. Updated daily.

## Calendar data

CALENDAR DATA		
Dataset	Notebooks	Description
Public Holidays	Azure Notebooks Azure Databricks	Worldwide public holiday data, covering 41 countries or regions from 1970 to 2099. Includes country and whether most people have paid time off.

## Access to datasets

With an Azure account, you can access open datasets using code or through the Azure service interface. The data is colocated with Azure cloud compute resources for use in your machine learning solution.

Open Datasets are available through the Azure Machine Learning UI and SDK. Open Datasets also provides Azure Notebooks and Azure Databricks notebooks you can use to connect data to Azure Machine Learning and Azure Databricks. Datasets can also be accessed through a Python SDK.

However, you don't need an Azure account to access Open Datasets; you can access them from any Python environment with or without Spark.

# Project Bonsai

## What is Project Bonsai?

Microsoft Project Bonsai is a low-code AI platform that speeds AI-powered automation development and part of the Autonomous Systems suite from Microsoft.

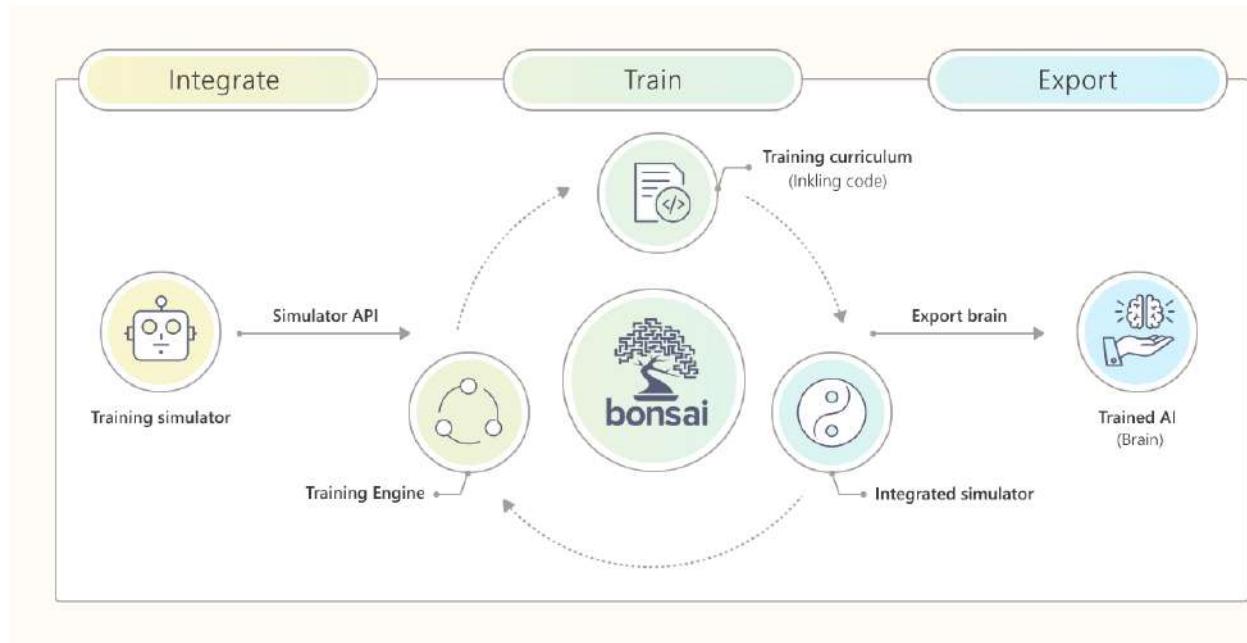
Use Bonsai to build AI components that can provide operator guidance or make independent decisions to optimize process variables, improve production efficiency, and reduce downtime.

The intuitive interface lets users create AI with their knowledge and experience without requiring additional resources. Bonsai gives you total control over how AI supports your process and total visibility into why your AI makes decisions and recommendations.

No neural net design required!

## The Bonsai platform

The Bonsai platform simplifies machine teaching with deep reinforcement learning so you can train and deploy smarter autonomous systems:



Infographic of the Bonsai platform. Shows a simulator icon flowing into a circular graph that connects the training engine, training curriculum, and integrated simulation. Another arrow flows away from the circular graph to indicate an exported brain.

- **Integrate** training simulations that implement real-world problems and provide realistic feedback during training.
- **Train** adaptive brains with intuitive goals and learning objectives, real-time success assessments, and automatic versioning control.

- **Export** the optimized brain as a Linux container and deploy it on-premises, in the cloud, or at the edge.

## Project Bonsai definition

Bonsai enables you to codify what an AI should learn by using human-friendly statements such as "avoid" and "maximize." This means you don't have to explicitly define detailed reward and terminal functions.

Bonsai includes integrated support for popular simulation software packages such as Simulink, MATLAB, and AnyLogic. Your simulation developers can therefore build machine-teaching simulations by using familiar software.

AI training with Bonsai has several components that work together, including:

- Training simulation
- Training engine
- Training curriculum
- Brains

# Training simulations for Bonsai

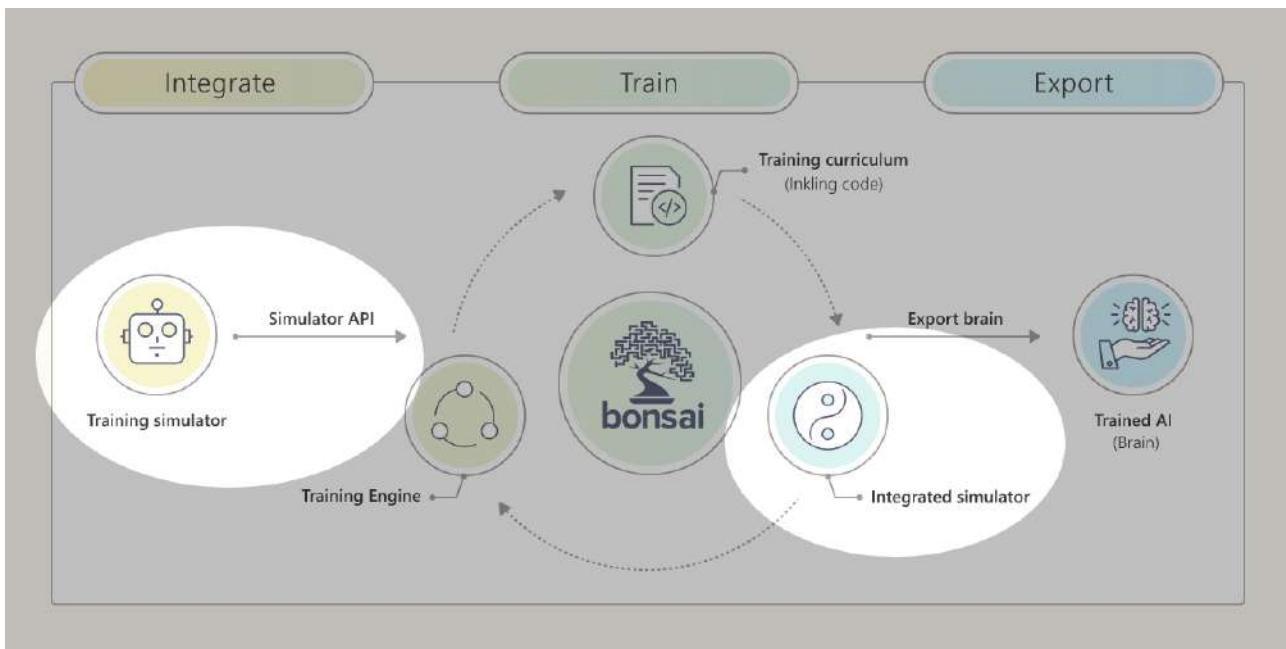
Training simulations replicate real-world systems to provide an authentic training environment for Bonsai brains. You can add simulations from [popular simulator software solutions](#) or use the Simulator API to integrate custom simulators.

## Supported Software

- MATLAB Simulink
- AnyLogic
- VP Link
- Custom dockerized containers

## API Support

- REST
- Python



Training simulations model real-world processes and change state as the brain applies actions. Robotics, industrial automation, supply chain logistics, and structural engineering are all domains that use simulations to model the behavior of complex systems.

Bonsai uses simulations and Deep Reinforcement Learning (DRL) to train brains. Training tasks can be as simple as "keep this pole upright" or as complex as "learn to walk." Generally speaking, any simulation that has a defined start state, iterates over time, and responds to external actions can integrate with Bonsai. But simulations that work well with Bonsai have the following characteristics:

- An appropriate level of fidelity so that strategies developed against the simulation are likely to work well in the real world.
- Useful visualization and data output while controlled by the brain for real-time assessment during training.
- A well-defined environment state that is accessible at each step of the simulation.
- A customizable start state so the brain can learn from a wide array of conditions.
- A set of discrete actions the brain can take to affect the state. For example: move a cart one step on a track, adjust a temperature by 1° Celsius.
- The ability to determine when the system gets into a state where further progress is impossible (a failure or invalid state). For example: the cart runs off the track, the current temperature exceeds a quality threshold.
- The ability to determine when the system reaches a success state. For example, a pole balances for a specific amount of time, the generated material passes QC requirements.

Determining the right level of fidelity for a simulation depends on:

- the precision required for individual actions.
- the probability that the AI could recover from an imperfect action in the real world.

For example, AI could compensate for an unexpectedly wide turn caused by a real-world car turning **1 km** faster than the simulated car the AI trained with. But, if that same car regularly turns **10 km** faster than the simulated car, the car could flip over or run off the road.

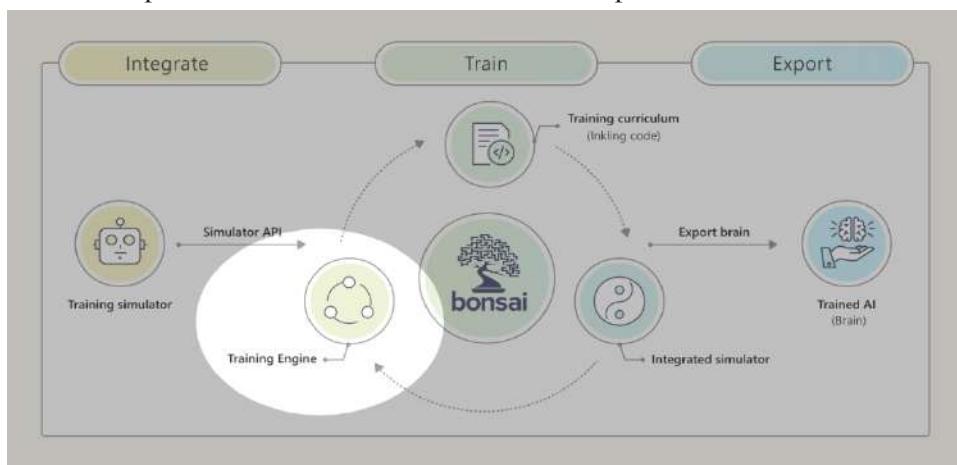
When considering your simulation approach, it may be helpful to look for people in your organization who have worked with simulation software before. Simulations originally created for other purposes can often be enhanced to work with the Bonsai training engine. Look for existing simulations with one or more of the following characteristics:

- Simulations used to train human operators.
- Simulations regularly used in conjunction with production systems.
- Simulations with well-defined benchmarks for accuracy and desired outcomes.

## Bonsai training engine

Bonsai has a training engine with four key components:

- **Architect**: generates learning models based on the training curriculum.
- **Instructor**: coordinates training for the Learner based on the curriculum and training data provided by the simulation.
- **Learner**: gains experience at solving the problem based on direction provided by the Instructor.
- **Predictor**: reports how the trained Learner will behave when presented with new data. The Predictor represents the trained brain that will be exported



Infographic of the Bonsai platform with the training engine highlighted.

## Architect

The Architect creates and optimizes learning topologies (neural networks) based on the training curriculum defined by the Inkling code. Essentially, the Architect does what a data scientist would do when evaluating the effectiveness of a neural network.

Based on the training curriculum and the available models, the Architect proposes the configuration of learning algorithms and topologies that have the best chance at learning the concepts in the model.

Currently, the Architect supports the following learning algorithms:

- Distributed Deep Q Network (APEX)
- Proximal Policy Optimization (PPO)
- Soft Actor Critic (SAC)

The set of heuristics the Architect uses is based on the same heuristics used routinely by the data science and machine learning experts who work on the Architect codebase.

## Instructor

The Instructor carries out the training plan by configuring the Learner and any data sources required based on the needs of the training curriculum. While some operations are batched, the Instructor is designed to work interactively. It responds in real time as the Learner iterates through the process of receiving data, computing a response, being assessed, and learning from the result.

## Learner

The Learner carries out the underlying AI algorithms selected by the Architect. During training, the Learner coordinates with the Instructor to set the starting parameters of the learning algorithm then determines a response and grades its performance.

In a deployed Brain, the Learner is responsible for instantiating the trained system and executing its computation when needed.

## Predictor

The Predictor is essentially a trained brain. Once trained, the AI algorithm is hosted in **prediction mode**. Prediction mode holds a brain for use as an HTTP API endpoint so that programmers can send input data to the brain to get back a prediction.

## Goals and objectives

Use your subject matter expertise to break complex problems into key objectives for the AI to learn. Then use Inkling to encode those objectives as goals and objectives.

Goals are a high-level specification of what you want the system to learn. They encapsulate your intentions for the AI without having to craft reward functions and early termination definitions.

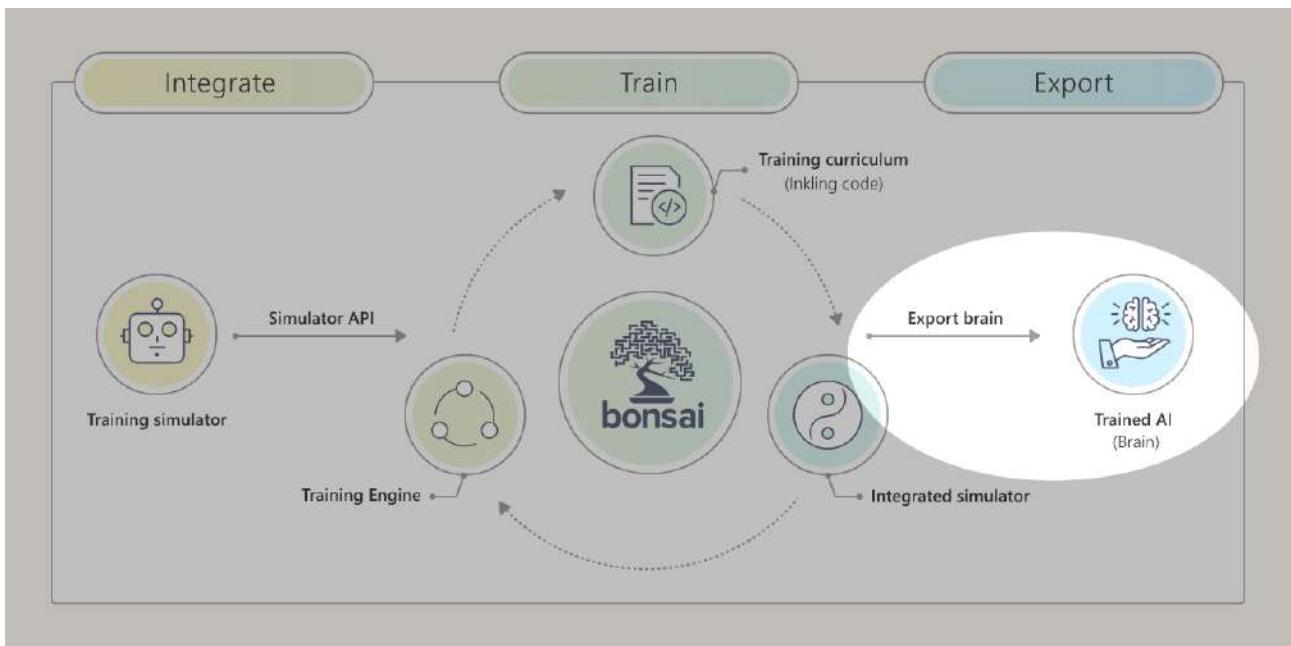
The training engine uses your goals to select the best learning algorithm for your brain then reports on training progress based on the goals you specified.

Available goal objectives include:

- **avoid:** Avoid a defined region.
- **drive:** Get to a target as quickly as possible and stay near the target.
- **maximize:** Push a target value as high as possible within a given range.
- **minimize:** Push a target value as low as possible within a given range.
- **reach:** Get to a target as quickly as possible.

## Bonsai brains

Bonsai brains are trained AI models with the ability to intelligently control and optimize real-world systems. Brains use deep reinforcement learning and, if desired, past results to resolve complex and conflicting goals.



You can generate a new brain version at any time, but Bonsai also provides automatic versioning any time you make significant changes to your training curriculum.

Compare training results, Inkling code, or the predictions of your latest version with previous versions then export the best one as a Linux container for use outside the platform.

## How to use Bonsai to deliver AI-powered automation

The Bonsai platform helps simplify machine teaching so you can train and deploy smarter autonomous systems. Bonsai enables you to:

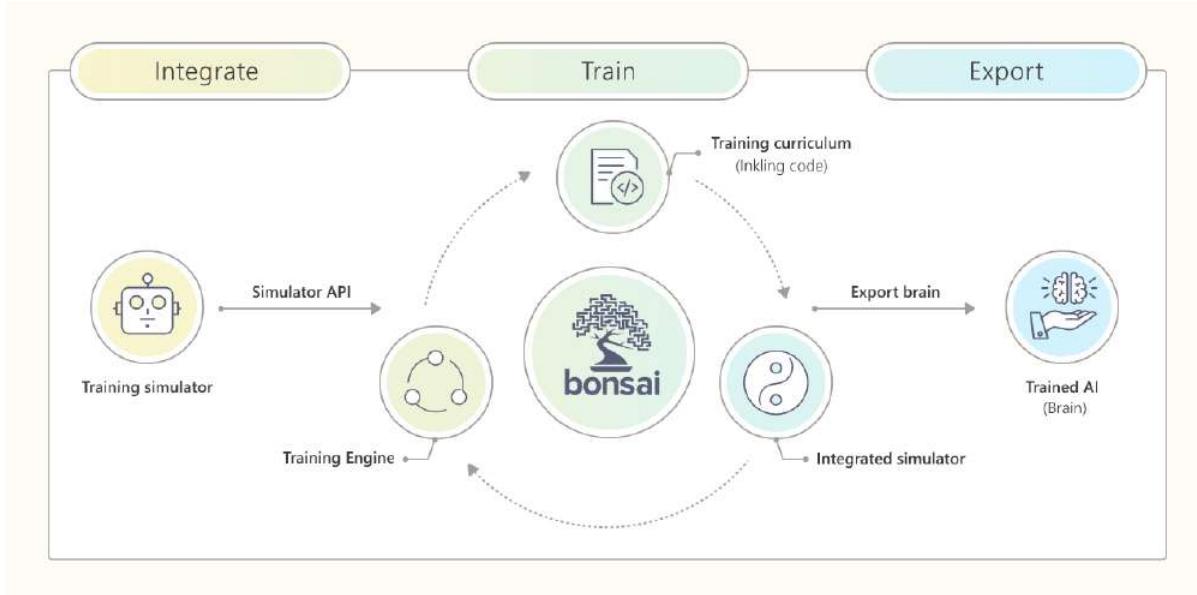
- Integrate training simulations that implement real-world problems.
- Provide realistic feedback during training.
- Train adaptive brains with:
  - Intuitive goals and learning objectives.
  - Real-time success assessments.
  - Automatic versioning control.
- Export the optimized brain and deploy it:
  - In the cloud.
  - At the edge.
  - On-premises.

## How Microsoft Project Bonsai works

In this unit, you'll learn how the Project Bonsai components work together. If you're a user of automation systems, you'll learn how Bonsai can streamline the process of automation development. As we've discussed, Bonsai consists of the components that the following table describes and the subsequent graphic displays:

### **HOW MICROSOFT PROJECT BONSAI WORKS**

<b><i>Component</i></b>	<b><i>Description</i></b>
<i>Simulation connector</i>	Connects a simulator to Bonsai.
<i>AI training engine</i>	Manages AI learning.
<i>Training curriculum (goals and</i>	Defines what the AI should learn using user-friendly terms.
<i>Brain exporter</i>	Packages the trained AI in a container for deployment in
<i>Inkling</i>	Is a customer programming language.



## How do these components work together?

In a typical scenario, you start by identifying a problem you want to solve by using AI. You can then take the following steps:

1. Create a simulation that models the real-world environment in which the AI will operate.
2. Test and verify the simulation locally.
3. Use a connector to import the simulation as a managed simulation package in Bonsai.
4. Design a teaching plan in Inkling using goals and objectives.
5. Iteratively train and assess the Bonsai brain's performance.
6. Export the fully trained AI as a Docker container. This occurs only after you determine the brain is trained appropriately.
7. Run verification against the exported AI.
8. Deploy the Docker container and integrate it with your real-world control system. This occurs only when you're satisfied with the AI performance.

## Work with simulations

Simulations replicate your real-world systems. This enables you to provide a valid training context in which your Bonsai brains can learn. When planning simulations, consider that they'll work well with Bonsai if they have the following characteristics:

- A suitable level of fidelity. This helps ensure that strategies you develop against the simulation are likely to work well in the real world.
- Useful visualization and data output. This helps ensure meaningful real-time assessment during training.
- A well-defined environment state. This must be accessible at each step of the simulation.

- A customizable start state. This helps ensure that the brain learns from various conditions.
- A set of discrete actions the brain can take to affect the state. For example:
  - Move a cart one step on a production line.
  - Adjust a temperature in a manufacturing process by 1° Celsius.
- The ability to identify system failure. An example is when the system gets into a state where further progress is impossible, such as:
  - A cart stops on the production line.
  - The temperature exceeds a manufacturing threshold.
- The ability to determine when the system reaches a success state. For example:
  - A cart moves down the line over a defined distance.
  - The temperature is maintained at a constant value for a defined period.

When considering your simulation approach, it might be helpful to enlist people in your organization who've worked with simulation software before.

## Tip

You can often repurpose and enhance existing simulations so they work with the Bonsai training engine. Try to locate simulations that:

- Are used to train human operators.
- Are used in conjunction with production systems.
- Have well-defined benchmarks for accuracy and desired outcomes.

## Work with the training engine

The training engine manages AI learning for Bonsai brains based on the training curriculum you define in your Inkling code. The training engine has four components:

- Architect
- Instructor
- Learner
- Predictor

## Architect

The Architect defines a training process for the Bonsai brain that will give it the best chance at learning the requested concepts. The Architect:

- creates a starting learning topology (neural networks).
- proposes an optimal configuration for the associated learning algorithms.
- optimizes the topology and algorithm configuration as the brain trains.

Essentially, the Architect does what a data scientist would do when evaluating the effectiveness of a neural network for training.

The Bonsai Architect currently supports the following learning algorithms (heuristics):

- Distributed Deep Q Network (APEX).
- Proximal Policy Optimization (PPO).
- Soft Actor Critic (SAC).

## Note

The current set of supported heuristics is based on heuristics used by the data-science and machine-learning experts who work on the Architect codebase.

## Instructor

The Instructor directs the learning process for the Bonsai brain during training. The Instructor:

- Configures the Learner
- Configures any required data sources.
- Works interactively with the Learner and responds in real time as training progresses.

## Learner

The Learner carries out the underlying AI algorithms that the Architect selects. During training, the Learner coordinates with the Instructor to set the learning algorithm's starting parameters, determines a response, and grades its performance.

In a deployed brain, the Learner must instantiate the trained system and execute its computation when needed.

## Predictor

The Predictor is the part of a trained brain that computes responses. It is called a predictor because it essentially "predicts" the right course of action based on past training.

## Work with the training curriculum

You specify instruction for the training engine as Inkling goals. Goals let you define your training curriculum with intuitive objectives instead of detailed reward and terminal functions. Available goal objectives include:

- **Avoid.** Do not enter a defined region.
- **Drive.** Get to a target as quickly as possible and stay near it.
- **Maximize.** Push a target value as high as possible within a given range.
- **Minimize.** Push a target value as low as possible within a given range.
- **Reach.** Get to a target as quickly as possible.

# When to use Microsoft Project Bonsai

In this unit, you'll learn how to determine whether Project Bonsai is a good fit for your AI-powered control problem. The criteria to consider are:

- Business value
- Available expertise
- Simulation overhead
- Business value
- Can you meaningfully optimize the performance of your current automation systems?

Determining the business value of additional optimization is crucial. Developing an AI solution with Bonsai makes the process **easier**, but not **easy**. You should make sure the time and resource investment is worth the benefits AI optimization will provide. Many organizations estimate a 2 percent to 5 percent increase in a key performance indicator (KPI) or increases in revenue or cost savings of roughly 1 million USD.

## Available expertise

- **Will you have access to the subject matter experts who understand the control system?**

Subject matter experts are crucial to successful AI development with Bonsai. The Inkling code (teaching curriculum) for your Bonsai brain is built on learning from subject matter experts and training will build additional optimization on top of that experience.

- **Does your organization include data scientists or AI experts?**

Bonsai is designed to make AI development accessible to those without AI experience. However, having data scientists or other AI experts available can help in cases where the problem requires advanced usage of the platform. For example, using explicit reward and terminal functions instead of goals or fine-tuning the learning algorithm.

## Simulation overhead

- **Do you have an existing, machine-teaching compatible simulation of your automation systems?**

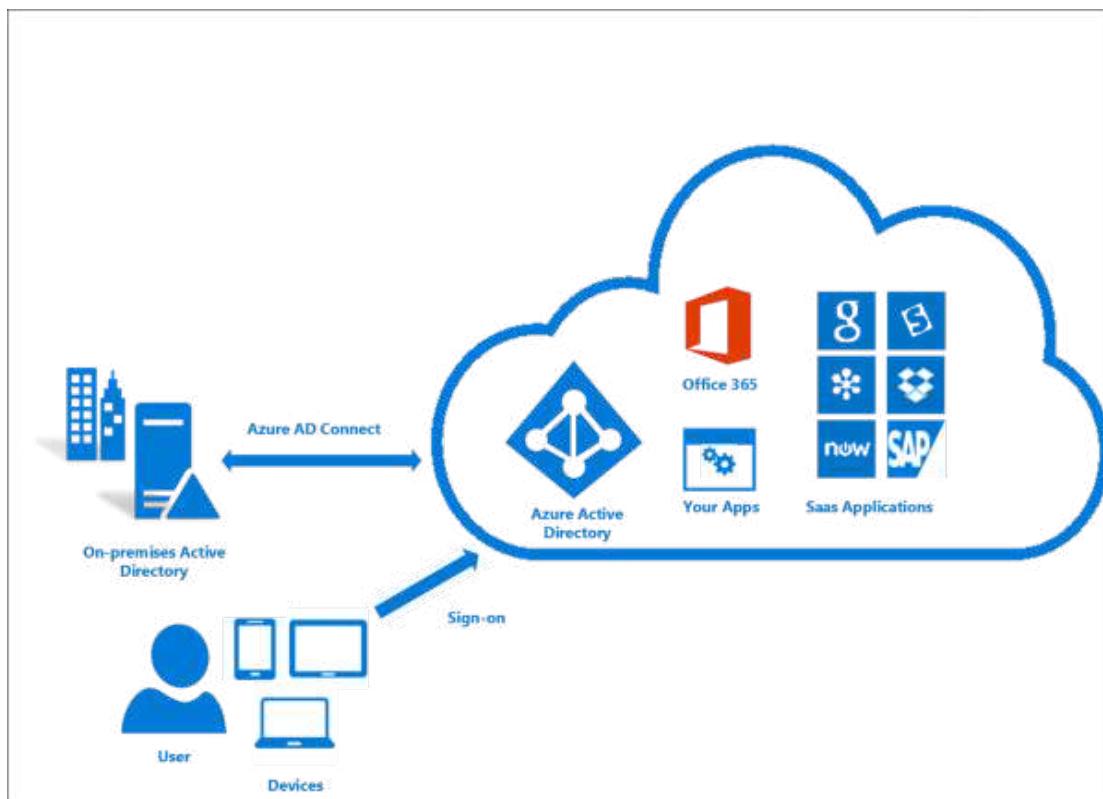
You cannot train a Bonsai brain without providing a simulation in which it can learn. But not all simulations are appropriate for machine teaching. It is important to take stock of any existing simulation assets you may already have access to, and whether your organization has the necessary expertise to modify an existing simulation model to make it machine teaching compatible or develop a new one from scratch.

# AZURE ACTIVE DIRECTORY



## What is Azure Active Directory (AAD)?

Azure Active Directory (Azure AD) is Microsoft's enterprise cloud-based identity and access management (IAM) solution. Azure AD is the backbone of the Office 365 system, and it can sync with on-premise Active Directory and provide authentication to other cloud-based systems via OAuth.



## Who uses Azure AD?

- IT admins. As an IT admin, you can use Azure AD to control access to your apps and your app resources, based on your business requirements.
- App developers. As an app developer, you can use Azure AD as a standards-based approach for adding single sign-on (SSO) to your app, allowing it to work with a user's pre-existing credentials.
- Microsoft 365, Office 365, Azure, or Dynamics CRM Online subscribers. As a subscriber, you're already using Azure AD. Each Microsoft 365, Office 365, Azure, and Dynamics CRM Online tenant is automatically an Azure AD tenant. You can immediately start to manage access to your integrated cloud apps.

## What are the Azure AD licenses?

Azure Active Directory Free. Provides user and group management, on-premises directory synchronization, basic reports, self-service password change for cloud users, and single sign-on across Azure, Microsoft 365, and many popular SaaS apps.

Azure Active Directory Premium P1. In addition to the Free features, P1 also lets your hybrid users access both on-premises and cloud resources.

Azure Active Directory Premium P2. In addition to the Free and P1 features, P2 also offers Azure Active Directory Identity Protection to help provide risk-based Conditional Access to your apps and critical company data and Privileged Identity Management to help discover, restrict, and monitor administrators and their access to resources and to provide just-in-time access when needed.

"Pay as you go" feature licenses. You can also get additional feature licenses, such as Azure Active Directory Business-to-Customer (B2C).

## Which features work in Azure AD?

After you choose your Azure AD license, you'll get access to some or all of the following features for your organization:

Category	Description
Application management	Manage your cloud and on-premises apps using Application Proxy, single sign-on, the My Apps portal (also known as the Access panel), and Software as a Service (SaaS) apps.
Authentication	Manage Azure Active Directory self-service password reset, Multi-Factor Authentication, custom banned password list, and smart lockout.
Azure Active Directory for developers	Build apps that sign in all Microsoft identities, get tokens to call Microsoft Graph, other Microsoft APIs, or custom APIs.
Business-to-Business (B2B)	Manage your guest users and external partners, while maintaining control over your own corporate data.
Business-to-Customer (B2C)	Customize and control how users sign up, sign in, and manage their profiles when using your apps.
Conditional Access	Manage access to your cloud apps.
Device Management	Manage how your cloud or on-premises devices access your corporate data.
Domain services	Join Azure virtual machines to a domain without using domain controllers.
Enterprise users	Manage license assignment, access to apps, and set up delegates using groups and administrator roles.

---

Hybrid identity	Use Azure Active Directory Connect and Connect Health to provide a single user identity for authentication and authorization to all resources, regardless of location (cloud or on-premises).
Identity governance	Manage your organization's identity through employee, business partner, vendor, service, and app access controls. You can also perform access reviews.
Identity protection	Detect potential vulnerabilities affecting your organization's identities, configure policies to respond to suspicious actions, and then take appropriate action to resolve them.
Managed identities for Azure resources	Provides your Azure services with an automatically managed identity in Azure AD that can authenticate any Azure AD-supported authentication service, including Key Vault.
Privileged identity management (PIM)	Manage, control, and monitor access within your organization. This feature includes access to resources in Azure AD and Azure, and other Microsoft Online Services, like Microsoft 365 or Intune.
Reports and monitoring	Gain insights into the security and usage patterns in your environment.

## Compare Active Directory to Azure Active Directory

Azure Active Directory is the next evolution of identity and access management solutions for the cloud. Microsoft introduced Active Directory Domain Services in Windows 2000 to give organizations the ability to manage multiple on-premises infrastructure components and systems using a single identity per user.

Azure AD takes this approach to the next level by providing organizations with an Identity as a Service (IDaaS) solution for all their apps across cloud and on-premises.

Most IT administrators are familiar with Active Directory Domain Services concepts. The following table outlines the differences and similarities between Active Directory concepts and Azure Active Directory.

Concept	Active Directory (AD)	Azure Active Directory
<b>Users</b>		
Provisioning: users	Organizations create internal users manually or use an in-house or automated provisioning system, such as the Microsoft Identity Manager, to integrate with an HR system.	Existing AD organizations use Azure AD Connect to sync identities to the cloud. Azure AD adds support to automatically create users from cloud HR systems. Azure AD can provision identities in SCIM enabled SaaS apps to automatically provide apps with the necessary details to allow access for users.

---

Provisioning: external identities	Organizations create external users manually as regular users in a dedicated external AD forest, resulting in administration overhead to manage the lifecycle of external identities (guest users)	Azure AD provides a special class of identity to support external identities. Azure AD B2B will manage the link to the external user identity to make sure they are valid.
Entitlement management and groups	Administrators make users members of groups. App and resource owners then give groups access to apps or resources.	Groups are also available in Azure AD and administrators can also use groups to grant permissions to resources. In Azure AD, administrators can assign membership to groups manually or use a query to dynamically include users to a group.  Administrators can use Entitlement management in Azure AD to give users access to a collection of apps and resources using workflows and, if necessary, time-based criteria.

---

---

Admin management	Organizations will use a combination of domains, organizational units, and groups in AD to delegate administrative rights to manage the directory and resources it controls.	Azure AD provides built-in roles with its Azure AD role-based access control (Azure AD RBAC) system, with limited support for creating custom roles to delegate privileged access to the identity system, the apps, and resources it controls. Managing roles can be enhanced with Privileged Identity Management (PIM) to provide just-in-time, time-restricted, or workflow-based access to privileged roles.
Credential management	Credentials in Active Directory are based on passwords, certificate authentication, and smartcard authentication. Passwords are managed using password policies that are based on password length, expiry, and complexity.	Azure AD uses intelligent password protection for cloud and on-premises. Protection includes smart lockout plus blocking common and custom password phrases and substitutions. Azure AD significantly boosts security through Multi-factor authentication and passwordless technologies, like FIDO2. Azure AD reduces support costs by providing users a self-service password reset system.

---

Apps

---



---

---

Infrastructure apps	Active Directory forms the basis for many infrastructure on-premises components, for example, DNS, DHCP, IPSec, WiFi, NPS, and VPN access	In a new cloud world, Azure AD, is the new control plane for accessing apps versus relying on networking controls. When users authenticate, Conditional access (CA), will control which users will have access to which apps under required conditions.
Traditional and legacy apps	Most on-premises apps use LDAP, Windows-Integrated Authentication (NTLM and Kerberos), or Header-based authentication to control access to users.	Azure AD can provide access to these types of on-premises apps using Azure AD application proxy agents running on-premises. Using this method Azure AD can authenticate Active Directory users on-premises using Kerberos while you migrate or need to coexist with legacy apps.
SaaS apps	Active Directory doesn't support SaaS apps natively and requires a federation system, such as AD FS.	SaaS apps supporting OAuth2, SAML, and WS-* authentication can be integrated to use Azure AD for authentication.

---

Line of business (LOB) apps with modern authentication	Organizations can use AD FS with Active Directory to support LOB apps requiring modern authentication.	LOB apps requiring modern authentication can be configured to use Azure AD for authentication.
Mid-tier/ Daemon services	Services running in on-premises environments normally use AD service accounts or group Managed Service Accounts (gMSA) to run.  These apps will then inherit the permissions of the service account.	Azure AD provides managed identities to run other workloads in the cloud. The lifecycle of these identities is managed by Azure AD and is tied to the resource provider that can't be used for other purposes to gain backdoor access.
<b>Devices</b>		
Mobile	Active Directory doesn't natively support mobile devices without third-party solutions.	Microsoft's mobile device management solution, Microsoft Intune, is integrated with Azure AD. Microsoft Intune provides device state information to the identity system to evaluate during authentication.

---

Windows desktops	Active Directory provides the ability to domain join Windows devices to manage them using Group Policy, System Center Configuration Manager, or other third-party solutions.	Windows devices can be joined to Azure AD. Conditional access can check if a device is Azure AD joined as part of the authentication process. Windows devices can also be managed with Microsoft Intune. In this case, conditional access will consider whether a device is compliant (for example, up-to-date security patches and virus signatures) before allowing access to the apps.
Windows servers	Active Directory provides strong management capabilities for on-premises Windows servers using Group Policy or other management solutions.	Windows servers virtual machines in Azure can be managed with Azure AD Domain Services. Managed identities can be used when VMs need access to the identity system directory or resources.
Linux/Unix workloads	Active Directory doesn't natively support non-Windows without third-party solutions, although Linux machines can be configured to authenticate with Active Directory as a Kerberos realm.	Linux/Unix VMs can use managed identities to access the identity system or resources. Some organizations migrate these workloads to cloud container technologies, which can also use managed identities.

---

## How Does Azure Active Directory Work?



Azure AD is a new system that Microsoft designed from the ground up to support cloud infrastructure. Azure AD uses REST APIs to pass data from one system to other cloud applications and systems that support REST (which is most cloud applications).

Unlike Windows AD, Azure AD is a flat structure in a single tenant. Think of the tenant as a circle that surrounds all your stuff. You can control the stuff inside the tenant, but once it leaves that circle you lose some agency over what happens to your stuff.

## Users and Groups

Users and groups are the basic building blocks for Azure AD. You can further organize users into groups that will all behave similarly.

DISPLAY NAME	USER NAME	SOURCED FROM
John Doe	johndoe@cloudalloc.com	Windows Azure Active Directory
Rick Rainey	rickrain@hotmail.com	Microsoft account

Figure 2: Azure AD Users sources

## Adding User and Groups to Azure AD

There are several methods to populate your users and groups in Azure AD.

- Use Azure AD Connect to sync users from Windows AD to Azure AD. Most enterprises that already have Windows AD use this method.
- You can create users manually in the Azure AD Management Portal.

- You can script the process to add new users with [PowerShell](#).
- Or you could program the process with the Azure AD Graph API.

## Common Attacks Against Azure AD

Azure AD is available from the internet, so it's a relatively easy target. A good password policy and multi-factor authentication, as well as behavioral monitoring of login activity and geo-hopping, can thwart most brute force attacks. Most. You still need to monitor your data to detect malicious activity inside your tenant in the event an attacker succeeds with a single login attempt.



Phishing is the other top attack we see against Azure AD users. Phishing can lead to credential theft or malware infection, which can provide attackers with a foothold to access your tenant. One of the better enhancements Azure AD provides is warnings when you open an email from an outsider or untrusted source.

## Azure AD B2C

### What is Azure Active Directory B2C?

Azure Active Directory B2C provides business-to-customer identity as a service. Azure AD B2C is a customer identity access management (CIAM) solution capable of supporting millions of users and billions of authentications per day. It takes care of the

scaling and safety of the authentication platform, monitoring, and automatically handling threats like denial-of-service, password spray, or brute force attacks.

Azure AD B2C is a separate service from Azure Active Directory (Azure AD). It is built on the same technology as Azure AD but for a different purpose. It allows businesses to build customer facing applications, and then allow anyone to sign up into those applications with no restrictions on user accounts.



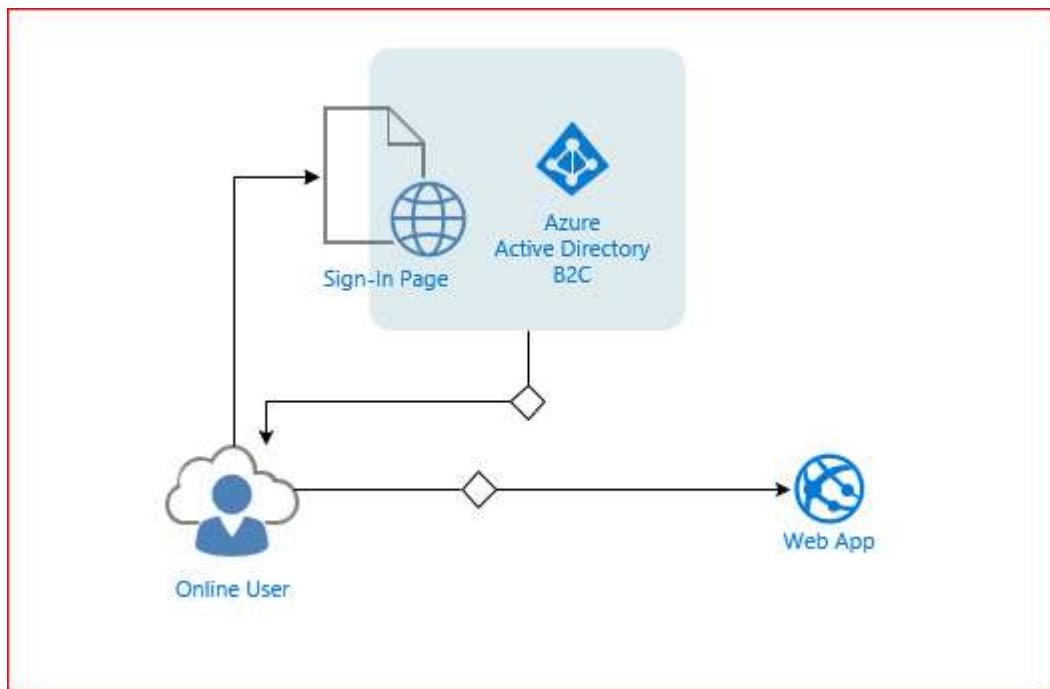
## Who uses Azure AD B2C?

Any business or individual who wishes to authenticate end users to their web/mobile applications using a white-label authentication solution. Apart from authentication, Azure AD B2C service is used for authorization such as access to API resources by authenticated users. Azure AD B2C is meant to be used by IT administrators and developers.

## Custom-branded identity solution

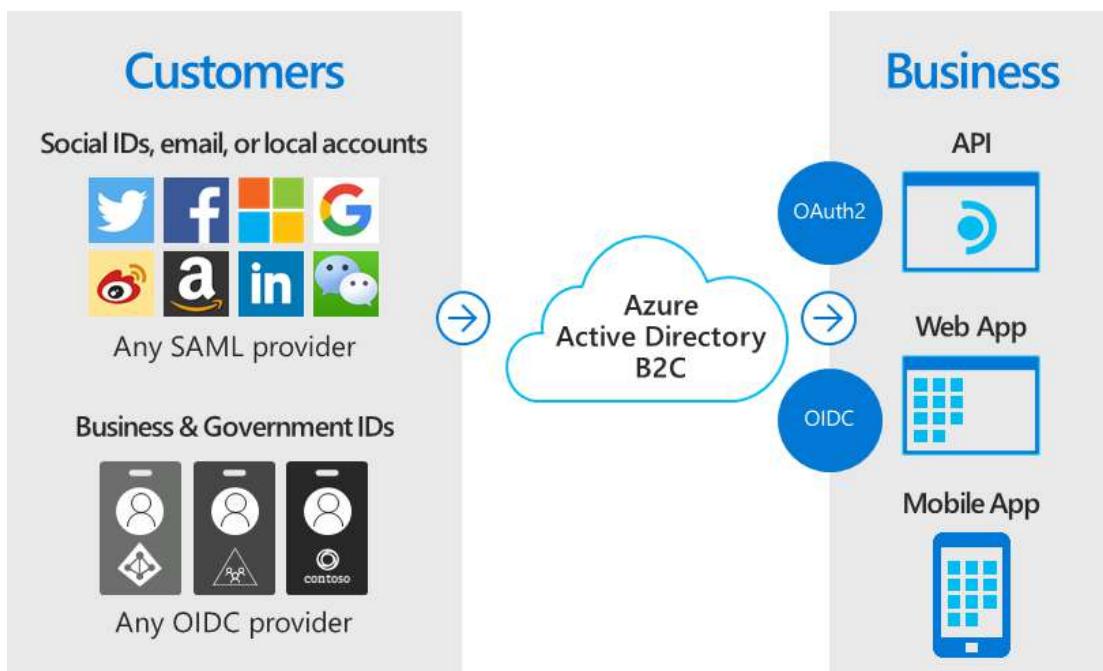
Azure AD B2C is a white-label authentication solution. You can customize the entire user experience with your brand so that it blends seamlessly with your web and mobile applications.

Customize every page displayed by Azure AD B2C when your users sign up, sign in, and modify their profile information. Customize the HTML, CSS, and JavaScript in your user journeys so that the Azure AD B2C experience looks and feels like it's a native part of your application.



## Single sign-on access with a user-provided identity

Azure AD B2C uses standards-based authentication protocols including OpenID Connect, OAuth 2.0, and Security Assertion Markup Language (SAML). It integrates with most modern applications and commercial off-the-shelf software.

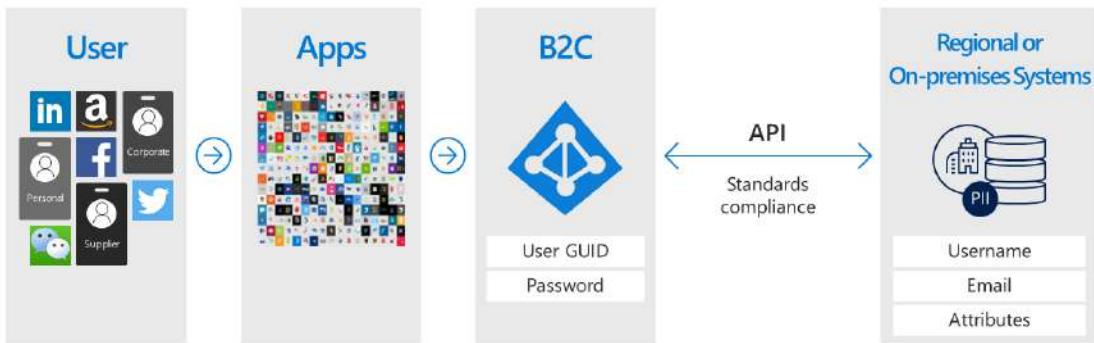


By serving as the central authentication authority for your web applications, mobile apps, and APIs, Azure AD B2C enables you to build a single sign-on (SSO) solution for them all. Centralize the collection of user profile and preference information, and capture detailed analytics about sign-in behavior and sign-up conversion.

## Integrate with external user stores

Azure AD B2C provides a directory that can hold 100 custom attributes per user. However, you can also integrate with external systems. For example, use Azure AD B2C for authentication, but delegate to an external customer relationship management (CRM) or customer loyalty database as the source of truth for customer data.

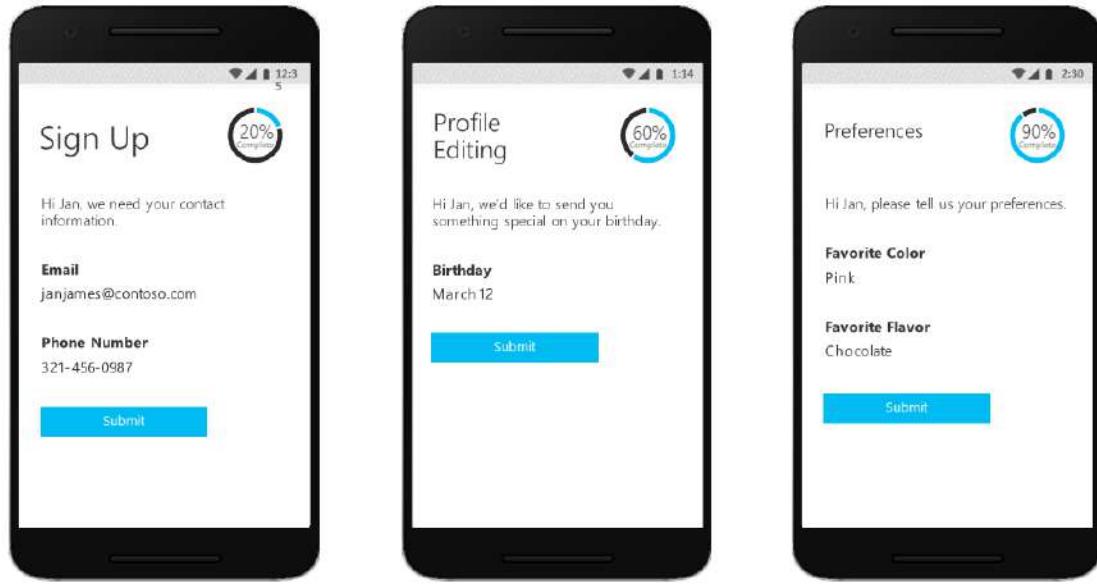
Another external user store scenario is to have Azure AD B2C handle the authentication for your application, but integrate with an external system that stores user profile or personal data. For example, to satisfy data residency requirements like regional or on-premises data storage policies. However, Azure AD B2C service itself is worldwide via the Azure public cloud.



Azure AD B2C can facilitate collecting the information from the user during registration or profile editing, then hand that data off to the external system via API. Then, during future authentications, Azure AD B2C can retrieve the data from the external system and, if needed, include it as a part of the authentication token response it sends to your application.

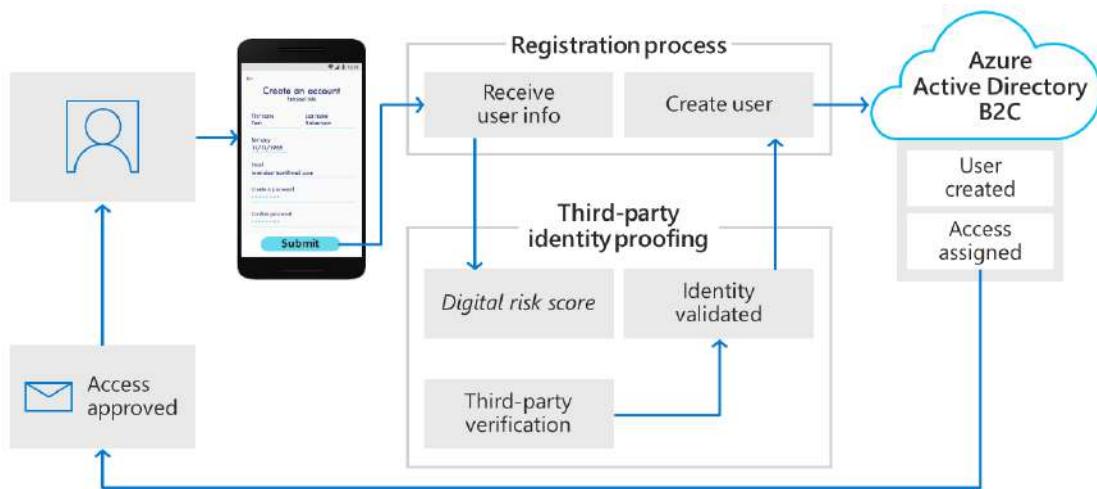
## Progressive profiling

Another user journey option includes progressive profiling. Progressive profiling allows your customers to quickly complete their first transaction by collecting a minimal amount of information. Then, gradually collect more profile data from the customer on future sign-ins.



## Third-party identity verification and proofing

Use Azure AD B2C to facilitate identity verification and proofing by collecting user data, then passing it to a third-party system to perform validation, trust scoring, and approval for user account creation.



The basic steps to setup Azure AD B2C are:

1. Create Azure AD B2C tenant.
2. Switch to Azure AD B2C directory.
3. Register your application(s).

4. Set up with any third-party identity providers.
5. Create sign-up, sign-in, password reset, and profile editing policies.
6. Configure your app to use the Azure AD B2C policies you created. This can be done using a Microsoft library for .NET or NodeJS web apps. Otherwise, you must use an OAuth 2.0 or OpenID Connect SDK (these are the two protocols Azure AD B2C uses).
7. (Optional) Create a custom user interface (UI) using HTML and CSS stylesheets.

## Azure AD B2C tenant

In Azure Active Directory B2C (Azure AD B2C), a tenant represents your organization and is a directory of users. Each Azure AD B2C tenant is distinct and separate from other Azure AD B2C tenants. An Azure AD B2C tenant is different from an Azure Active Directory tenant, which you may already have.

The primary resources you work with in an Azure AD B2C tenant are:

- Directory - The directory is where Azure AD B2C stores your users' credentials, profile data, and your application registrations.
- Application registrations - Register your web, mobile, and native applications with Azure AD B2C to enable identity management. You can also register any APIs you want to protect with Azure AD B2C.
- User flows and custom policies - Create identity experiences for your applications with built-in user flows and fully configurable custom policies:
  - User flows help you quickly enable common identity tasks like sign-up, sign-in, and profile editing.
  - Custom policies let you build complex identity workflows unique to your organization, customers, employees, partners, and citizens.
- Sign-in options - Azure AD B2C offers various sign-up and sign-in options for users of your applications:
  - Username, email, and phone sign-in - Configure your Azure AD B2C local accounts to allow sign-up and sign-in with a username, email address, phone number, or a combination of methods.
  - Social identity providers - Federate with social providers like Facebook, LinkedIn, or Twitter.
  - External identity providers - Federate with standard identity protocols like OAuth 2.0, OpenID Connect, and more.
- Keys - Add and manage encryption keys for signing and validating tokens, client secrets, certificates, and passwords.

## Accounts in Azure AD B2C

Azure AD B2C defines several types of user accounts. Azure Active Directory, Azure Active Directory B2B, and Azure Active Directory B2C share these account types.

- Work account - Users with work accounts can manage resources in a tenant, and with an administrator role, can also manage tenants. Users with work accounts can create new consumer accounts, reset passwords, block/unblock accounts, and set permissions or assign an account to a security group.
- Guest account - External users you invite to your tenant as guests. A typical scenario for inviting a guest user to your Azure AD B2C tenant is to share administration responsibilities.
- Consumer account - Accounts that are managed by Azure AD B2C user flows and custom policies.

NAME	USER NAME	USER TYPE	SOURCE
Colby Marble	hmosley@contoso.com	Member	LinkedIn
Harriett Mosley	hmosley@contoso.com	Member	Azure Active Directory
Julio Reuter	q*****@gmail.com	Member	Google
Katina Knowles	kknowles@contoso.com	Member	Azure Active Directory
Kelly Shields	kshields@fourthcoffee.com	Member	Facebook
Rigoberto Dugas		Member	LinkedIn
Roslyn Fry		Member	Federated Azure Active Directory
Warren Boatright	wboatright@contoso.com	Member	Azure Active Directory
Young Underwood	z*****@outlook.com	Member	Microsoft Account

Figure: User directory within an Azure AD B2C tenant in the Azure portal

## Consumer accounts

With a consumer account, users can sign in to the applications that you've secured with Azure AD B2C. Users with consumer accounts can't, however, access Azure resources, for example the Azure portal.

A consumer account can be associated with these identity types:

- Local identity, with the username and password stored locally in the Azure AD B2C directory. We often refer to these identities as "local accounts."
- Social or enterprise identities, where the identity of the user is managed by a federated identity provider. For example, Facebook, Google, Microsoft, ADFS, or Salesforce.

A user with a consumer account can sign in with multiple identities. For example username, email, employee ID, government ID, and others. A single account can have multiple identities, both local and social.

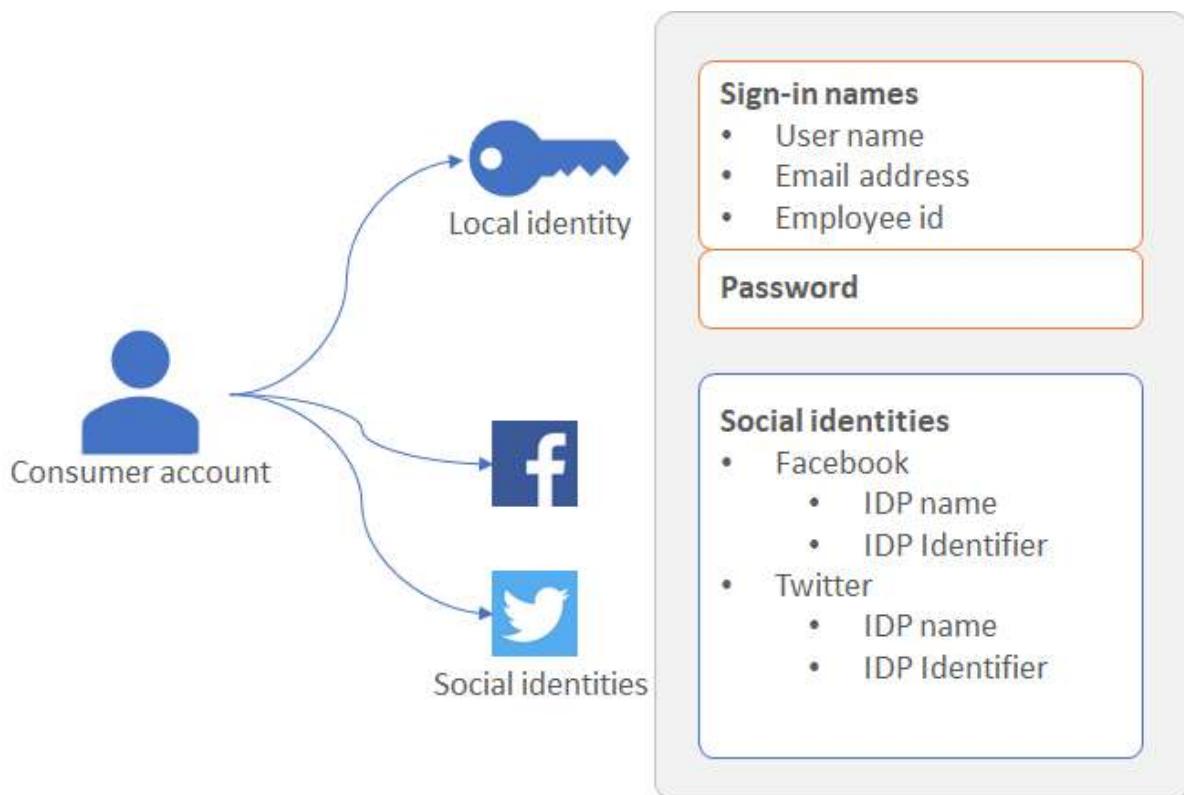


Figure: A single consumer account with multiple identities in Azure AD B2C

## Local account sign-in options

Azure AD B2C provides various ways in which users can authenticate a user. Users can sign-in to a local account, by using username and password, phone verification (also known as password-less authentication). Email sign-up is enabled by default in your local account identity provider settings.

## User profile attributes

Azure AD B2C lets you manage common attributes of consumer account profiles. For example display name, surname, given name, city, and others.

You can also extend the Azure AD schema to store additional information about your users. For example, their country/region of residency, preferred language, and preferences like whether they want to subscribe to a newsletter or enable multifactor authentication.

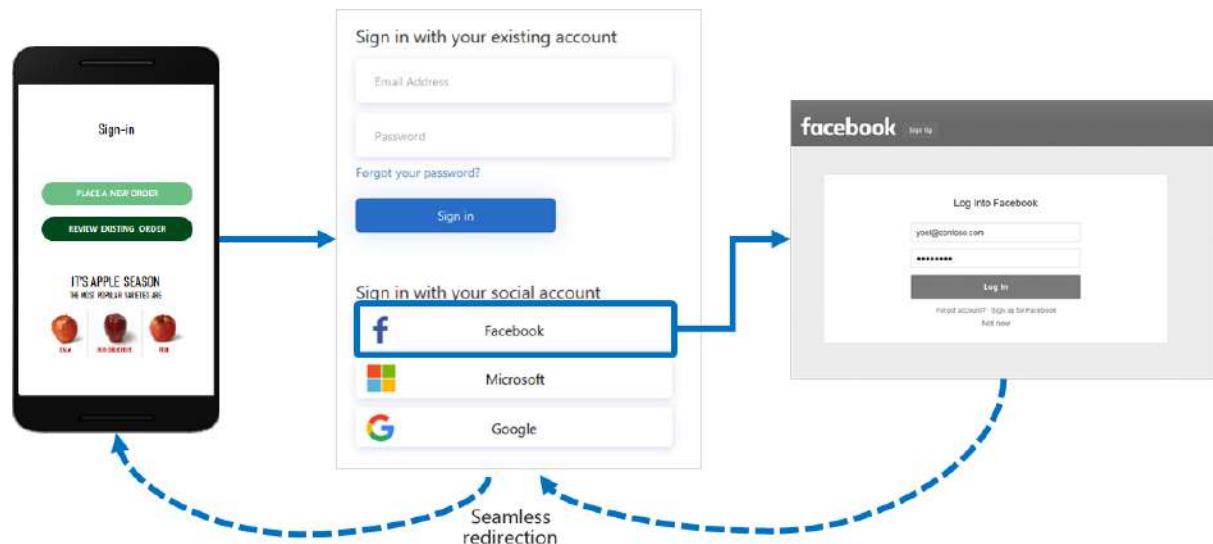
## Sign-in with external identity providers

You can configure Azure AD B2C to allow users to sign in to your application with credentials from social and enterprise identity providers. Azure AD B2C can federate with identity providers that support OAuth 1.0, OAuth 2.0, OpenID Connect, and SAML protocols. For example, Facebook, Microsoft account, Google, Twitter, and AD-FS.



With external identity provider federation, you can offer your consumers the ability to sign in with their existing social or enterprise accounts, without having to create a new account just for your application.

On the sign-up or sign-in page, Azure AD B2C presents a list of external identity providers the user can choose for sign-in. Once they select one of the external identity providers, they're taken (redirected) to the selected provider's website to complete the sign in process. After the user successfully signs in, they're returned to Azure AD B2C for authentication of the account in your application.



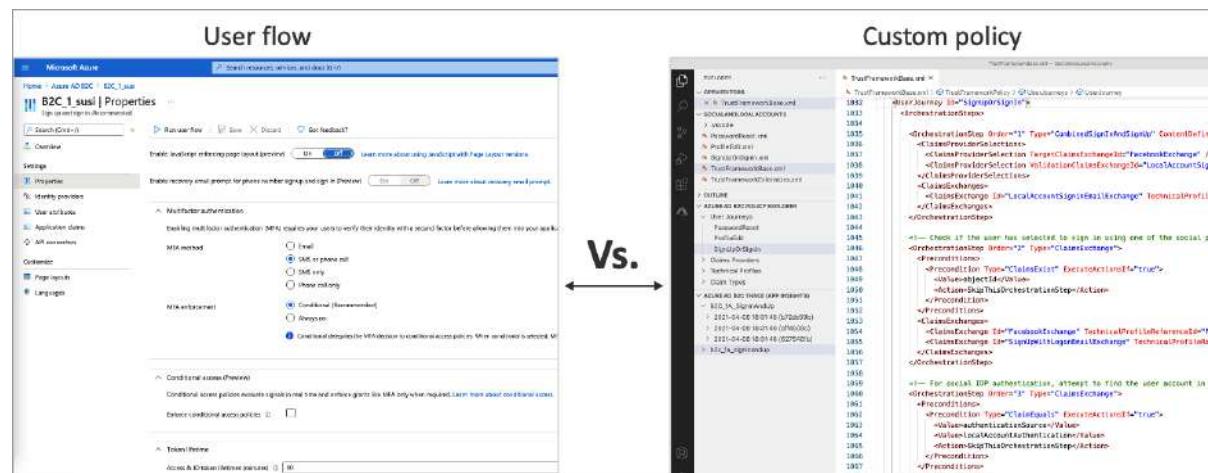
## Identity experiences: User-flows or Custom Policies

In Azure AD B2C, you can define the business logic that users follow to gain access to your application. For example, you can determine the sequence of steps users follow when they sign in, sign up, edit a profile, or reset a password. After completing the sequence, the user acquires a token and gains access to your application.

In Azure AD B2C, there are two ways to provide identity user experiences:

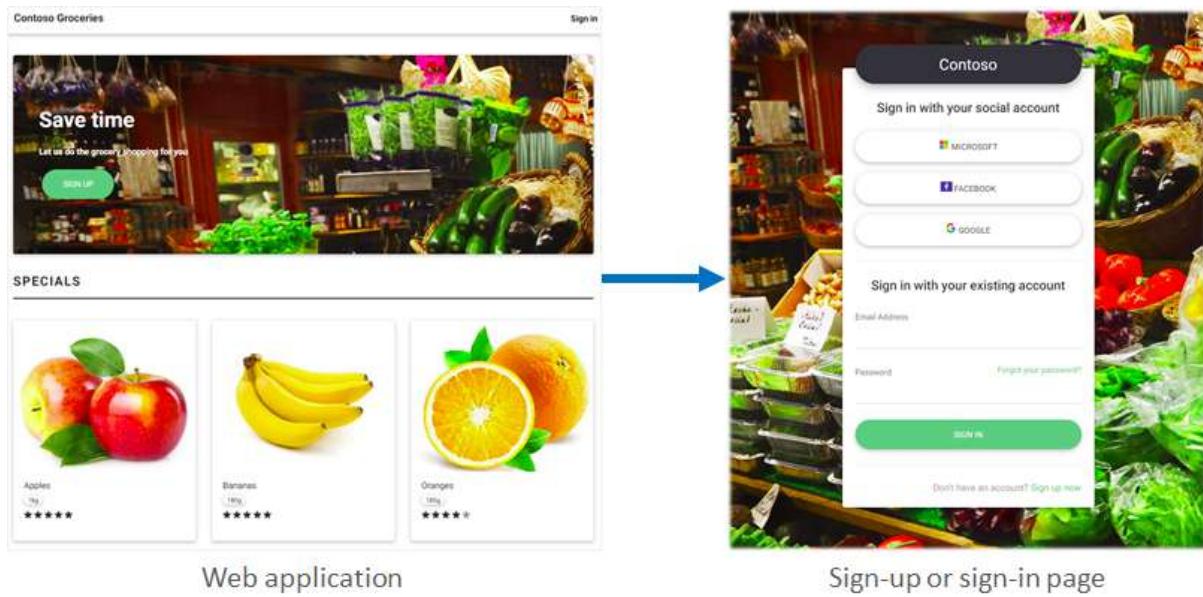
- User flows are predefined, built-in, configurable policies that we provide so you can create sign-up, sign-in, and policy editing experiences in minutes.
  - Custom policies enable you to create your own user journeys for complex identity experience scenarios.

The following screenshot shows the user flow settings UI, versus custom policy configuration files.



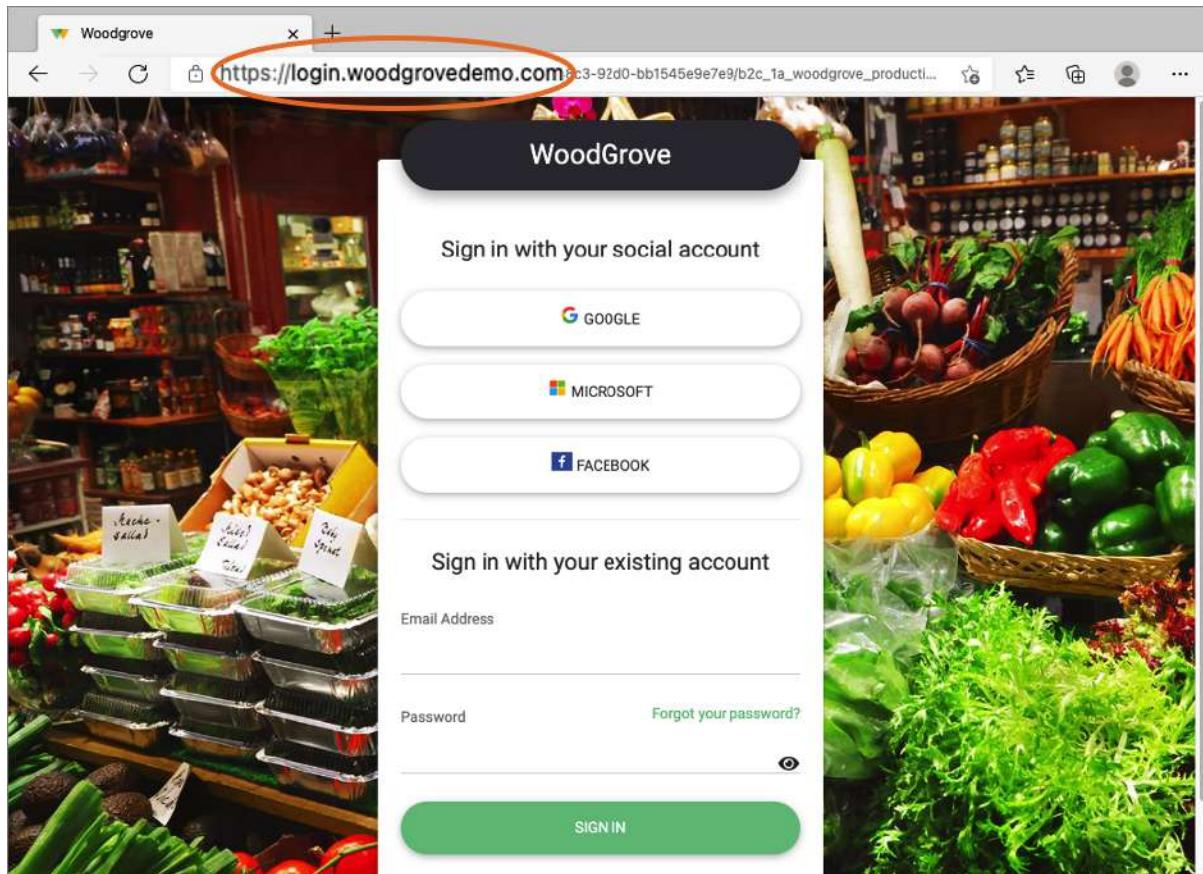
# User interface

In Azure AD B2C, you can craft your users' identity experiences so that the pages that are shown blend seamlessly with the look and feel of your brand. You get nearly full control of the HTML and CSS content presented to your users when they proceed through your application's identity journeys. With this flexibility, you can maintain brand and visual consistency between your application and Azure AD B2C.



## Custom domain

You can customize your Azure AD B2C domain in the redirect URLs for Azure AD B2C. Custom domain allows you to create a seamless experience so that the pages that are shown blend seamlessly with the domain name of your application.



From the user's perspective, they remain in your domain during the sign-in process rather than redirecting to the Azure AD B2C default domain .b2clogin.com.

## Localization

Language customization in Azure AD B2C allows you to accommodate different languages to suit your customer needs. Microsoft provides the translations for 36 languages, but you can also provide your own translations for any language. Even if your experience is provided for only a single language, you can customize any text on the pages.

The image displays three identical sign-in forms side-by-side, each in a different language: English, Spanish, and Hindi. The English version has labels like "Sign in with your existing account" and "Sign in with your social account". The Spanish version has labels like "Iniciar sesión con su cuenta existente" and "Iniciar sesión con su cuenta social". The Hindi version has labels like "अपने मौजूदा खाते के साथ साइन इन करें" and "अपने सोशल खाते के साथ साइन इन करें". All three versions include fields for "Email Address" and "Password", a "Forgot your password?" link, a blue "Sign in" button, and social login buttons for Microsoft, Google, and Facebook.

## Email verification

Azure AD B2C ensures valid email addresses by requiring customers to verify them during the sign-up, and password reset flows. It also prevents malicious actors from using automated processes to generate fraudulent accounts in your applications.

The image shows three sequential steps in the sign-up process. Step 1: A user enters their email address (emily@contoso.com) and clicks the "Send verification code" button. Step 2: An email from Microsoft arrives, titled "WoodGrove Groceries account email verification code". It contains a verification code: 687821, which is highlighted with a red circle. Step 3: The user enters the verification code (687821) into the "Verification Code" field on a second sign-up form. The form also includes fields for "Email Address", "New Password", "Confirm New Password", "Display Name", "Given Name", and "Surname", along with "Create" and "Cancel" buttons.

You can customize the email to users that sign up to use your applications. By using the third-party email provider, you can use your own email template and From: address and subject, as well as support localization and custom one-time password (OTP) settings.

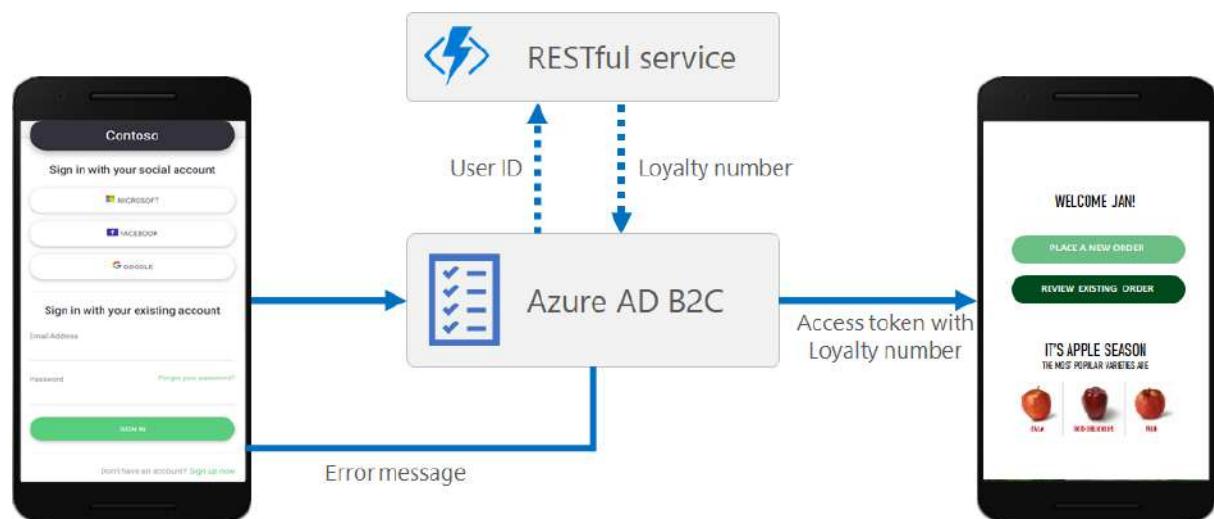
## Add your own business logic and call RESTful API

You can integrate with a RESTful API in both user flows and custom policies. The difference is, in user flows, you make calls at specific places, whereas in custom policies, you add your own business logic to the journey. This feature allows you to retrieve and use data from external identity sources. Azure AD B2C can exchange data with a RESTful service to:

- Display custom user-friendly error messages.
- Validate user input to prevent malformed data from persisting in your user directory. For example, you can modify the data entered by the user, such as capitalizing their first name if they entered it in all lowercase.
- Enrich user data by further integrating with your corporate line-of-business application.
- Using RESTful calls, you can send push notifications, update corporate databases, run a user migration process, manage permissions, audit databases, and more.

Loyalty programs are another scenario enabled by Azure AD B2C's support for calling REST APIs. For example, your RESTful service can receive a user's email address, query your customer database, then return the user's loyalty number to Azure AD B2C.

The return data can be stored in the user's directory account in Azure AD B2C. The data then can be further evaluated in subsequent steps in the policy, or be included in the access token.



You can add a REST API call at any step in the user journey defined by a custom policy. For example, you can call a REST API:

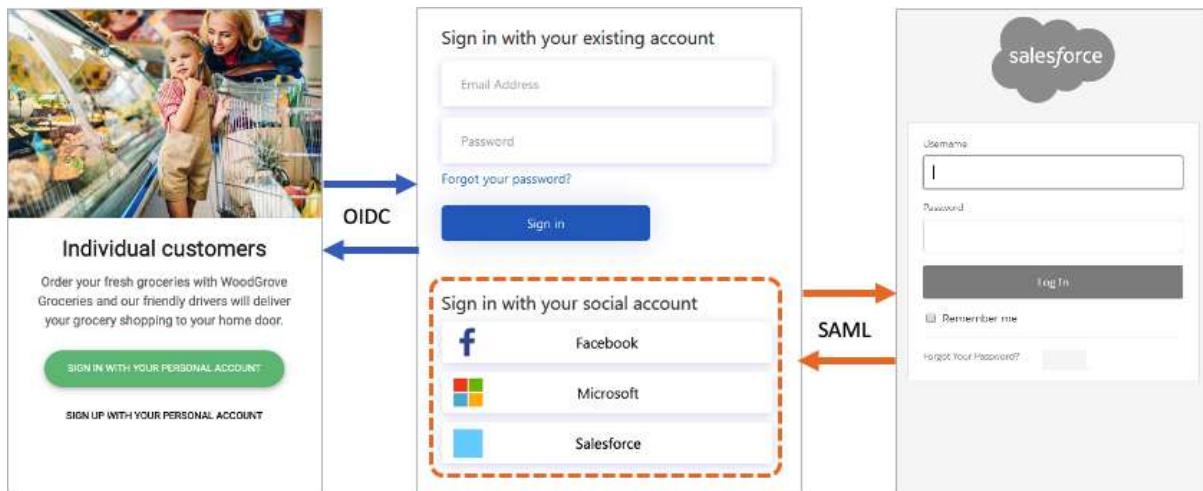
- During sign-in, just before Azure AD B2C validates the credentials
- Immediately after sign-in
- Before Azure AD B2C creates a new account in the directory
- After Azure AD B2C creates a new account in the directory
- Before Azure AD B2C issues an access token

## Protocols and tokens

For applications, Azure AD B2C supports the OAuth 2.0, OpenID Connect, and SAML protocols for user journeys. Your application starts the user journey by issuing authentication requests to Azure AD B2C. The result of a request to Azure AD B2C is a security token, such as an ID token, access token, or SAML token. This security token defines the user's identity within the application.

For external identities, Azure AD B2C supports federation with any OAuth 1.0, OAuth 2.0, OpenID Connect, and SAML identity providers.

The following diagram shows how Azure AD B2C can communicate using various protocols within the same authentication flow:



1. The relying party application starts an authorization request to Azure AD B2C using OpenID Connect.
2. When a user of the application chooses to sign in using an external identity provider that uses the SAML protocol, Azure AD B2C invokes the SAML protocol to communicate with that identity provider.

3. After the user completes the sign-in operation with the external identity provider, Azure AD B2C then returns the token to the relying party application using OpenID Connect.

## Application integration

When a user wants to sign in to your application, the application initiates an authorization request to a user flow- or custom policy-provided endpoint. The user flow or custom policy defines and controls the user's experience. When they complete a user flow, for example the sign-up or sign-in flow, Azure AD B2C generates a token, then redirects the user back to your application.



Multiple applications can use the same user flow or custom policy. A single application can use multiple user flows or custom policies. For example, to sign in to an application, the application uses the sign up or sign in user flow. After the user has signed in, they may want to edit their profile, so the application initiates another authorization request, this time using the profile edit user flow.

## Multi Factor authentication (MFA)

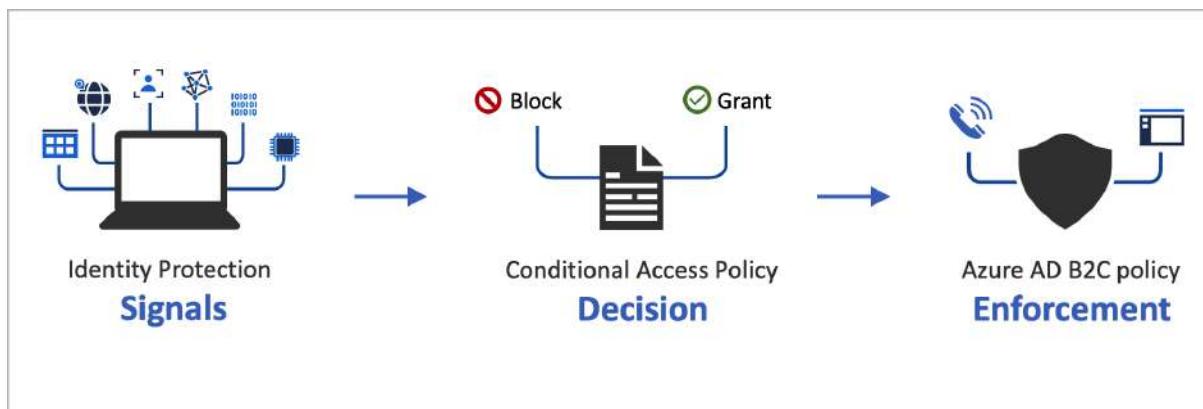
Azure AD B2C Multi-Factor Authentication (MFA) helps safeguard access to data and applications while maintaining simplicity for your users. It provides extra security by

requiring a second form of authentication, and delivers strong authentication by offering a range of easy-to-use authentication methods.

Your users may or may not be challenged for MFA based on configuration decisions that you can make as an administrator.

## Conditional Access

Azure AD Identity Protection risk-detection features, including risky users and risky sign-ins, are automatically detected and displayed in your Azure AD B2C tenant. You can create Conditional Access policies that use these risk detections to determine remediation actions and enforce organizational policies.



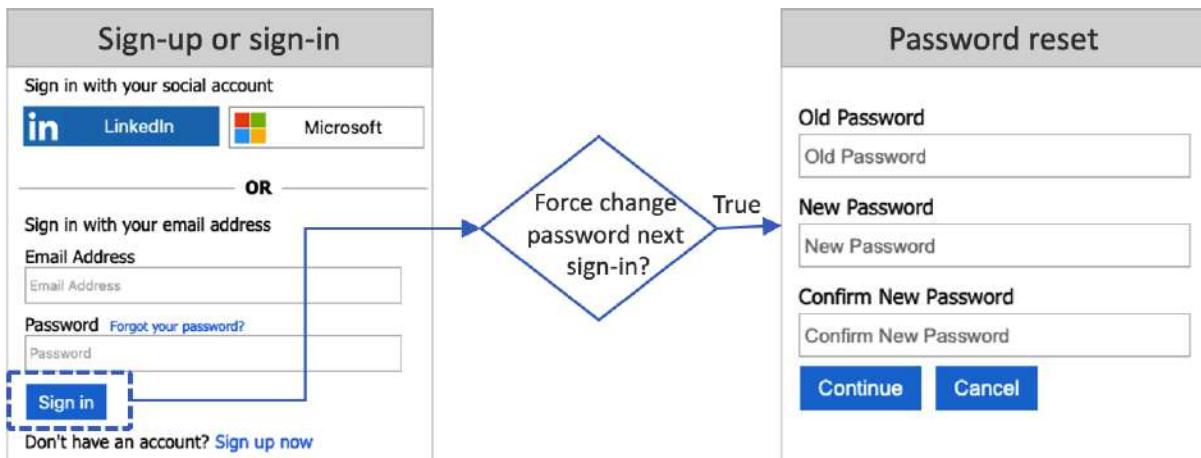
Azure AD B2C evaluates each sign-in event and ensures that all policy requirements are met before granting the user access. Risky users or sign-ins may be blocked, or challenged with a specific remediation like multifactor authentication (MFA).

## Password complexity

During sign up or password reset, your users must supply a password that meets complexity rules. By default, Azure AD B2C enforces a strong password policy. Azure AD B2C also provides configuration options for specifying the complexity requirements of the passwords your customers use.

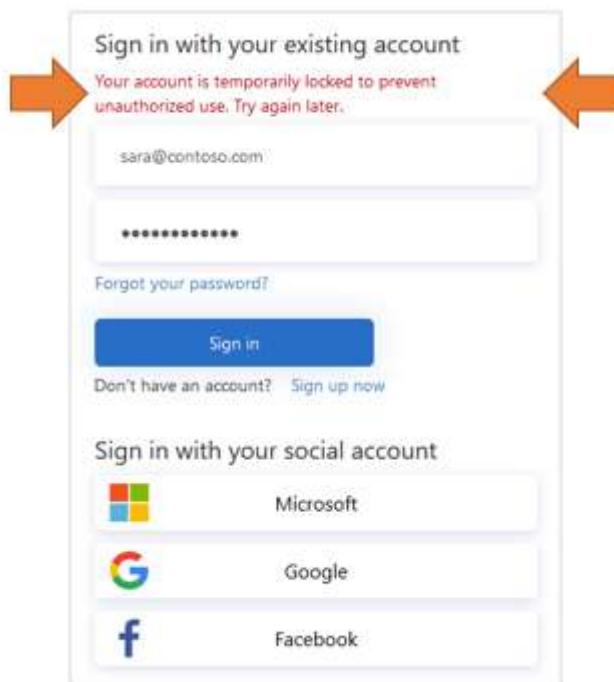
### Force password reset

As an Azure AD B2C tenant administrator, you can reset a user's password if the user forgets their password. Or you would like to force them to reset the password periodically.



## Smart account lockout

To prevent brute-force password guessing attempts, Azure AD B2C uses a sophisticated strategy to lock accounts based on the IP of the request, the passwords entered, and several other factors. The duration of the lockout is automatically increased based on risk and the number of attempts.

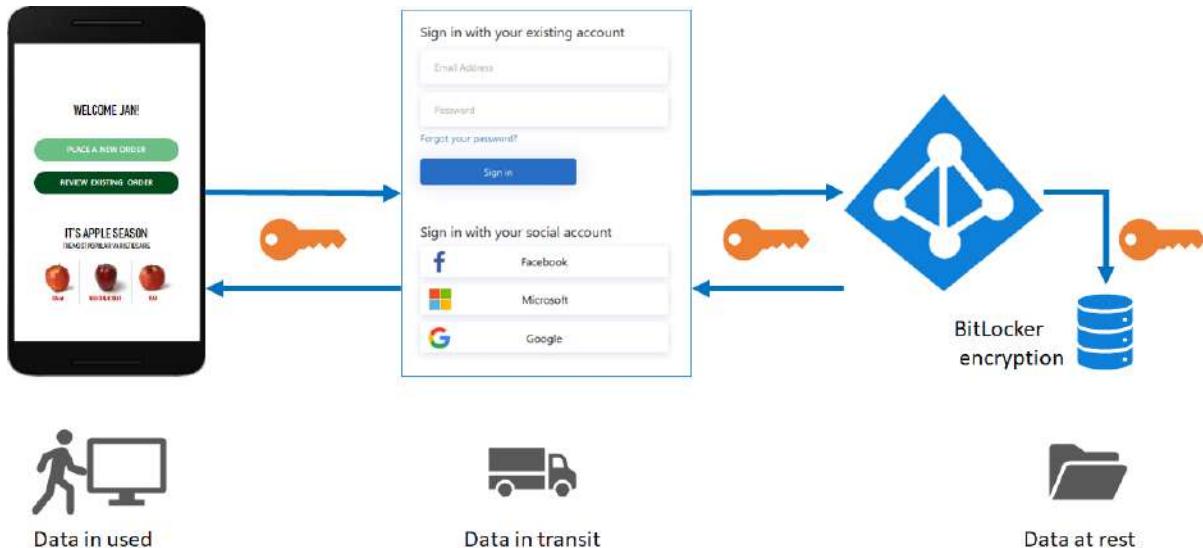


## Protect resources and customer identities

Azure AD B2C complies with the security, privacy, and other commitments described in the Microsoft Azure Trust Center.

Sessions are modeled as encrypted data, with the decryption key known only to the Azure AD B2C Security Token Service. A strong encryption algorithm, AES-192, is used. All

communication paths are protected with TLS for confidentiality and integrity. Our Security Token Service uses an Extended Validation (EV) certificate for TLS. In general, the Security Token Service mitigates cross-site scripting (XSS) attacks by not rendering untrusted input.



## Access to user data

Azure AD B2C tenants share many characteristics with enterprise Azure Active Directory tenants used for employees and partners. Shared aspects include mechanisms for viewing administrative roles, assigning roles, and auditing activities.

You can assign roles to control who can perform certain administrative actions in Azure AD B2C, including:

- Create and manage all aspects of user flows
- Create and manage the attribute schema available to all user flows
- Configure identity providers for use in direct federation
- Create and manage trust framework policies in the Identity Experience Framework (custom policies)
- Manage secrets for federation and encryption in the Identity Experience Framework (custom policies)

## Auditing and logs

Azure AD B2C emits audit logs containing activity information about its resources, issued tokens, and administrator access. You can use the audit logs to understand platform activity and diagnose issues. Audit log entries are available soon after the activity that generated the event occurs.

In an audit log, which is available for your Azure AD B2C tenant or for a particular user, you can find information including:

- Activities concerning the authorization of a user to access B2C resources (for example, an administrator accessing a list of B2C policies)
- Activities related to directory attributes retrieved when an administrator signs in using the Azure portal
- Create, read, update, and delete (CRUD) operations on B2C applications
- CRUD operations on keys stored in a B2C key container
- CRUD operations on B2C resources (for example, policies and identity providers)
- Validation of user credentials and token issuance

The screenshot shows the 'Audit logs' section for user 'Lisa Wood'. The left sidebar includes 'Manage' (Profile, Directory role, Groups, Applications, Licenses, Devices, Azure resources, Authentication methods), 'Activity' (Sign-ins, Audit logs - selected), 'Troubleshooting + Support' (Troubleshoot, New support request), and a 'Details' section. The main area has a 'Columns' dropdown, 'Refresh', and 'Download' buttons. Filter options include 'Service: All', 'Initiated By (Actor): Enter actor name or user', 'Category: All', 'Activity: All', 'Status: All', 'Target: Enter target name or user', 'Date: Last 7 days', 'Show dates as: Local (selected)', and 'Apply' and 'Reset' buttons. Below is a table of audit log entries:

DATE	SERVICE	CATEGORY	ACTIVITY	STATUS	TARGET(S)	INITIATED BY (ACTOR)
3/18/2019, 3:09:47 PM	B2C	Authentication	Verify phone number	Success	6ebce029-74bf-460f-8677... 0239a9cc-309c-4d41-87f1...	
3/18/2019, 3:09:31 PM	B2C	Authentication	Send SMS to verify phone number	Success	6ebce029-74bf-460f-8677... 0239a9cc-309c-4d41-87f1...	
3/18/2019, 3:09:25 PM	B2C	Authentication	Validate local account credentials	Success	a42005e0-72db-491d-a76a... 0239a9cc-309c-4d41-87f1...	
3/18/2019, 3:07:53 PM	B2C	Authentication	Validate local account credentials	Success	a42005e0-72db-491d-a76a... 0239a9cc-309c-4d41-87f1...	
3/18/2019, 3:07:54 PM	B2C	Authentication	Verify phone number	Success	6ebce029-74bf-460f-8677... 0239a9cc-309c-4d41-87f1...	
3/18/2019, 3:06:46 PM	B2C	Authentication	Send SMS to verify phone number	Success	6ebce029-74bf-460f-8677... 0239a9cc-309c-4d41-87f1...	

Below the table is a 'Details' section with tabs for 'Activity' (selected), 'Target(s)', and 'Modified Properties'. The 'Activity' tab shows:

ACTIVITY	INITIATED BY (ACTOR)	ADDITIONAL DETAILS
DATE: 3/18/2019, 3:09:25 PM	TYPE: Application	TenantId: sunflowerdemo.onmicrosoft.com
ACTIVITY TYPE: Validate local account credentials	DISPLAY NAME:	PolicyId: B2C_1_Profile_Edit
CORRELATION ID: 8174cfdf-df35-4bbe-b2d5-9d4e948c9e5c	APP ID: 2ed4b543-8ea7-40b7-9ba8-9c8f42aeeeda	ApplicationId: 0239a9cc-309c-4d41-87f1-31288fe2e282
CATEGORY: Authentication	SERVICE PRINCIPAL ID:	Client: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/80.0.3987.148 Safari/537.36
STATUS: Success	SERVICE PRINCIPAL NAME: 0239a9cc-309c-4d41-87f1-31288fe2e282	LocalAccountUsername: [REDACTED]
STATUS REASON: N/A		ClientIpAddress: [REDACTED]

## Usage analytics

Azure AD B2C allows you to discover when people sign up or sign in to your app, where the users are located, and what browsers and operating systems they use.

By integrating Azure Application Insights into Azure AD B2C custom policies, you can gain insight into how people sign up, sign in, reset their password or edit their profile. With such knowledge, you can make data-driven decisions for your upcoming development cycles.

## **Region availability and data residency**

Azure AD B2C service is generally available worldwide with the option for data residency in regions as specified in Products available by region. Data residency is determined by the country/region you select when you create your tenant.

## **Automation using Microsoft Graph API**

Use MS graph API to manage your Azure AD B2C directory. You can also create the Azure AD B2C directory itself. You can manage users, identity providers, user flows, custom policies and many more.

## **Supported Azure AD features**

An Azure AD B2C tenant is different than an Azure Active Directory tenant, which you may already have, but it relies on it. The following Azure AD features can be used in your Azure AD B2C tenant.

<b>Feature</b>	<b>Azure AD</b>	<b>Azure AD B2C</b>
Groups	Groups can be used to manage administrative and user accounts.	Groups can be used to manage administrative accounts. Consumer accounts can not be members of any group.
Inviting External Identities guests	You can invite guest users and configure External Identities features such as federation and sign-in with Facebook and Google accounts.	You can invite only a Microsoft account or an Azure AD user as a guest to your Azure AD tenant for accessing applications or managing tenants. For consumer accounts, you use Azure AD B2C user flows and custom policies to manage users and sign-up or sign-in with external identity providers, such as Google or Facebook.

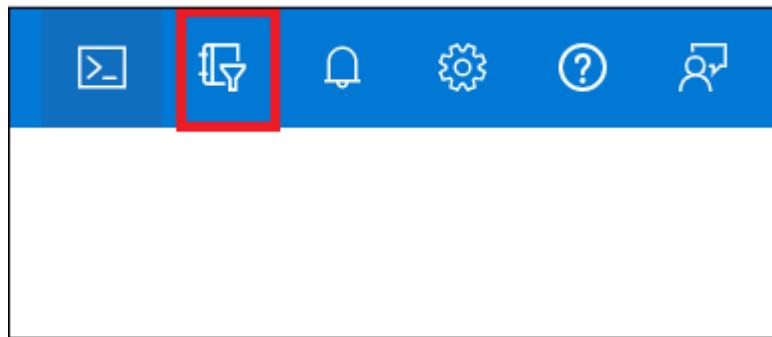
Roles and administrators	Fully supported for administrative and user accounts.	Roles are not supported with consumer accounts. Consumer accounts don't have access to any Azure resources.
Custom domain names	You can use Azure AD custom domains for administrative accounts only.	Consumer accounts can sign in with a username, phone number, or any email address. You can use custom domains in your redirect URLs.
Conditional Access	Fully supported for administrative and user accounts.	A subset of Azure AD Conditional Access features is supported with consumer accounts. Learn how to configure Azure AD B2C conditional access.
Premium P1	Fully supported for Azure AD premium P1 features. For example, Password Protection, Hybrid Identities, Conditional Access, Dynamic groups, and more.	A subset of Azure AD Conditional Access features is supported with consumer accounts. Learn how to configure Azure AD B2C Conditional Access.
Premium P2	Fully supported for Azure AD premium P2 features. For example, Identity Protection, and Identity Governance.	A subset of Azure AD Identity Protection features is supported with consumer accounts. Learn how to Investigate risk with Identity Protection and configure Azure AD B2C Conditional Access.

## Tutorial: Create an Azure Active Directory B2C tenant

Before your applications can interact with Azure Active Directory B2C (Azure AD B2C), they must be registered in a tenant that you manage.

### Create an Azure AD B2C tenant

1. Sign in to the Azure portal.
2. Switch to the directory that contains your subscription:
  - In the Azure portal toolbar, select the Directories + subscriptions filter icon.

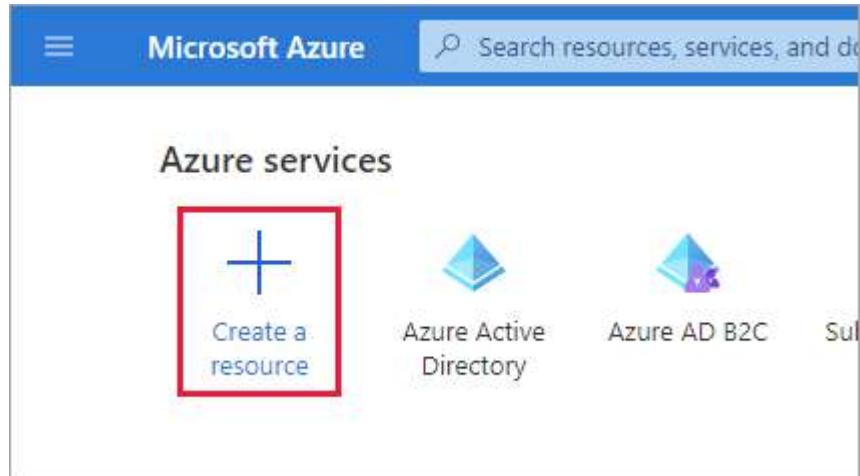


- Find the directory that contains your subscription and select the Switch button next to it. Switching a directory reloads the portal.

A screenshot of the Azure Portal settings page titled "Portal settings | Directories + subscriptions". The "Directories + subscriptions" section is selected in the left sidebar. On the right, there is a table listing directories. The first row, "Default Directory", is marked as "Current". A "Switch" button is located next to the "Current" status. The "Switch" button is highlighted with a red box.

3. Add Microsoft.AzureActiveDirectory as a resource provider for the Azure subscription you're using:
  - On the Azure portal, search for and select Subscriptions.
  - Select your subscription, and then in the left menu, select Resource providers. If you don't see the left menu, select the Show the menu for < name of your subscription > icon at the top left part of the page to expand it.
  - Make sure the Microsoft.AzureActiveDirectory row shows a status of Registered. If it doesn't, select the row, and then select Register.

4. On the Azure portal menu or from the Home page, select Create a resource.



5. Search for Azure Active Directory B2C, and then select Create.
6. Select Create a new Azure AD B2C Tenant.

The screenshot shows a navigation path: Home > New > Azure Active Directory B2C > Create new B2C Tenant or Link to existing Tenant. Below this, there are two main options: "Create a new Azure AD B2C Tenant" (with a purple user icon) and "Link an existing Azure AD B2C Tenant to my Azure subscription" (with a yellow key icon). Both options have a small informational icon (info symbol) next to them. The "Create a new Azure AD B2C Tenant" option is highlighted with a red rectangular border.

7. On the Create a directory page, enter the following:
  - Organization name - Enter a name for your Azure AD B2C tenant.
  - Initial domain name - Enter a domain name for your Azure AD B2C tenant.
  - Country or region - Select your country or region from the list. This selection can't be changed later.
  - Subscription - Select your subscription from the list.
  - Resource group - Select or search for the resource group that will contain the tenant.

Home > New > Azure Active Directory B2C > Create new B2C Tenant or Link to existing Tenant >

## Create a directory

Azure Active Directory

\* Basics \* Configuration Review + create

**Directory details**

Configure your new directory

Organization name \* ⓘ Contoso B2C ✓

Initial domain name \* ⓘ contosob2c ✓  
contosob2c.onmicrosoft.com

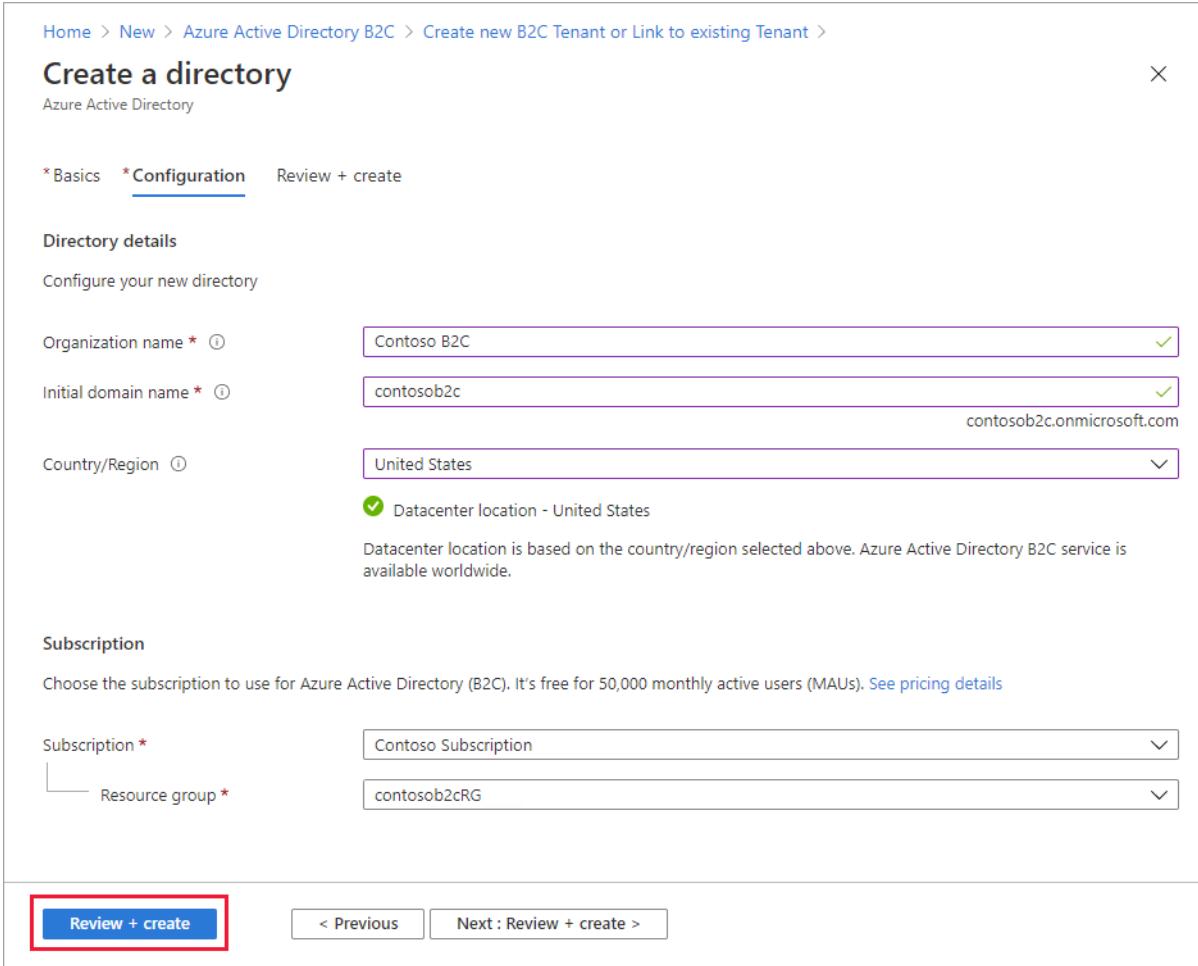
Country/Region ⓘ United States ✓  
Datacenter location - United States  
Datacenter location is based on the country/region selected above. Azure Active Directory B2C service is available worldwide.

**Subscription**

Choose the subscription to use for Azure Active Directory (B2C). It's free for 50,000 monthly active users (MAUs). [See pricing details](#)

Subscription \* Contoso Subscription ✓  
Resource group \* contosob2cRG ✓

**Review + create** < Previous Next : Review + create >



8.

9. Select Review + create.

10. Review your directory settings. Then select Create.

You can link multiple Azure AD B2C tenants to a single Azure subscription for billing purposes. To link a tenant, you must be an admin in the Azure AD B2C tenant and be assigned at least a Contributor role within the Azure subscription.

## Select your B2C tenant directory

To start using your new Azure AD B2C tenant, you need to switch to the directory that contains the tenant:

1. In the Azure portal toolbar, select the Directories + subscriptions filter icon.
2. On the All Directories tab, find the directory that contains your Azure AD B2C tenant and then select the Switch button next to it.

If at first you don't see your new Azure B2C tenant in the list, refresh your browser window or sign out and sign back in. Then in the Azure portal toolbar, select the Directories + subscriptions filter again.

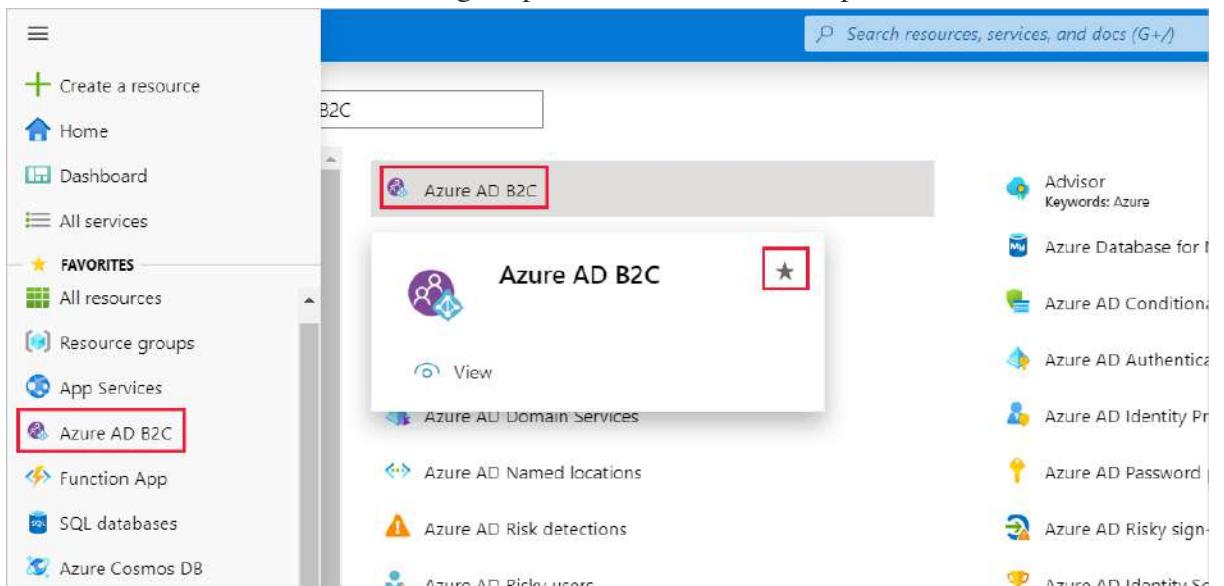
## Add Azure AD B2C as a favorite (optional)

This optional step makes it easier to select your Azure AD B2C tenant in the following and all subsequent tutorials.

Instead of searching for *Azure AD B2C* in All services every time you want to work with your tenant, you can instead favorite the resource. Then, you can select it from the portal menu's Favorites section to quickly browse to your Azure AD B2C tenant.

You only need to perform this operation once. Before performing these steps, make sure you've switched to the directory containing your Azure AD B2C tenant as described in the previous section, Select your B2C tenant directory.

1. Sign in to the Azure portal.
2. In the Azure portal menu, select All services.
3. In the All services search box, search for Azure AD B2C, hover over the search result, and then select the star icon in the tooltip. Azure AD B2C now appears in the Azure portal under Favorites.
4. If you want to change the position of your new favorite, go to the Azure portal menu, select Azure AD B2C, and then drag it up or down to the desired position.



## Tutorial: Register a web application in Azure Active Directory B2C

Before your applications can interact with Azure Active Directory B2C (Azure AD B2C), they must be registered in a tenant that you manage. This tutorial shows you how to register a web application using the Azure portal.

## Register a web application

To register a web application in your Azure AD B2C tenant, you can use our new unified App registrations experience or our legacy Applications (Legacy) experience.

- App registrations
  1. Sign in to the Azure portal.
  2. Make sure you're using the directory that contains your Azure AD B2C tenant. Select the Directories + subscriptions icon in the portal toolbar.
  3. On the Portal settings | Directories + subscriptions page, find your Azure AD B2C directory in the Directory name list, and then select Switch.
  4. In the Azure portal, search for and select Azure AD B2C.
  5. Select App registrations, and then select New registration.
  6. Enter a Name for the application. For example, webapp1.
  7. Under Supported account types, select Accounts in any identity provider or organizational directory (for authenticating users with user flows).
  8. Under Redirect URI, select Web, and then enter <https://jwt.ms> in the URL text box.

The redirect URI is the endpoint to which the user is sent by the authorization server (Azure AD B2C, in this case) after completing its interaction with the user, and to which an access token or authorization code is sent upon successful authorization. In a production application, it's typically a publicly accessible endpoint where your app is running, like <https://contoso.com/auth-response>. For testing purposes like this tutorial, you can set it to <https://jwt.ms>, a Microsoft-owned web application that displays the decoded contents of a token (the contents of the token never leave your browser). During app development, you might add the endpoint where your application listens locally, like <https://localhost:5000>. You can add and modify redirect URIs in your registered applications at any time.

The following restrictions apply to redirect URIs:

- The reply URL must begin with the scheme https.
  - The reply URL is case-sensitive. Its case must match the case of the URL path of your running application. For example, if your application includes as part of its path .../abc/response-oidc, do not specify .../ABC/response-oidc in the reply URL. Because the web browser treats paths as case-sensitive, cookies associated with .../abc/response-oidc may be excluded if redirected to the case-mismatched .../ABC/response-oidc URL.
9. Under Permissions, select the Grant admin consent to openid and offline\_access permissions check box.
  10. Select Register.

- Applications (Legacy)
  1. Sign in to the Azure portal.
  2. Make sure you're using the directory that contains your Azure AD B2C tenant. Select the Directories + subscriptions icon in the portal toolbar.
  3. On the Portal settings | Directories + subscriptions page, find your Azure AD B2C directory in the Directory name list, and then select Switch.
  4. In the Azure portal, search for and select Azure AD B2C.
  5. Select Applications (Legacy), and then select Add.
  6. Enter a name for the application. For example, webapp1.
  7. For Include web app/ web API, select Yes.
  8. For Reply URL, enter an endpoint where Azure AD B2C should return any tokens that your application requests. For example, you could set it to listen locally at `http://localhost:5000`. You can add and modify redirect URIs in your registered applications at any time.

The following restrictions apply to redirect URIs:

- The reply URL must begin with the scheme https, unless using localhost.
- The reply URL is case-sensitive. Its case must match the case of the URL path of your running application. For example, if your application includes as part of its path `.../abc/response-oidc`, do not specify `.../ABC/response-oidc` in the reply URL. Because the web browser treats paths as case-sensitive, cookies associated with `.../abc/response-oidc` may be excluded if redirected to the case-mismatched `.../ABC/response-oidc` URL.

9. Select Create to complete the application registration.

## Create a client secret

For a web application, you need to create an application secret. The client secret is also known as an *application password*. The secret will be used by your application to exchange an authorization code for an access token.

- App registrations
  1. In the Azure AD B2C - App registrations page, select the application you created, for example webapp1.
  2. In the left menu, under Manage, select Certificates & secrets.
  3. Select New client secret.
  4. Enter a description for the client secret in the Description box. For example, `clientsecret1`.
  5. Under Expires, select a duration for which the secret is valid, and then select Add.

6. Record the secret's Value for use in your client application code. This secret value is never displayed again after you leave this page. You use this value as the application secret in your application's code.
- Applications (Legacy)
  1. In the Azure AD B2C - Applications page, select the application you created, for example webapp1.
  2. Select Keys and then select Generate key.
  3. Select Save to view the key. Make note of the App key value. You use this value as the application secret in your application's code.

### **Enable ID token implicit grant**

The defining characteristic of the implicit grant is that tokens, such as ID and access tokens, are returned directly from Azure AD B2C to the application. For web apps, such as ASP.NET Core web apps and <https://jwt.ms>, that request an ID token directly from the authorization endpoint, enable the implicit grant flow in the app registration.

1. In the left menu, under Manage, select Authentication.
2. Under Implicit grant, select both the Access tokens and ID tokens check boxes.
3. Select Save.

## **Tutorial: Create user flows and custom policies in Azure Active Directory B2C**

### **Create a sign-up and sign-in user flow**

The sign-up and sign-in user flow handles both sign-up and sign-in experiences with a single configuration. Users of your application are led down the right path depending on the context.

1. Sign in to the Azure portal.
2. Make sure you're using the directory that contains your Azure AD B2C tenant. Select the Directories + subscriptions icon in the portal toolbar.
3. On the Portal settings | Directories + subscriptions page, find your Azure AD B2C directory in the Directory name list, and then select Switch.
4. In the Azure portal, search for and select Azure AD B2C.

5. Under Policies, select User flows, and then select New user flow.

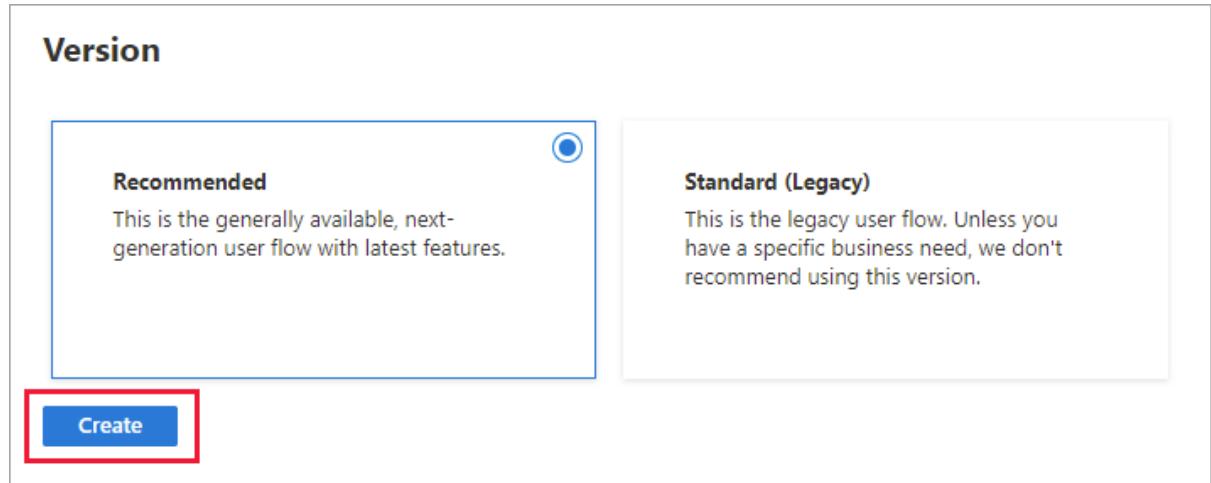
The screenshot shows the Azure AD B2C | User flows interface. On the left, there's a sidebar with links like Home, Overview, Manage (App registrations, Applications (Legacy), Identity providers, Company Branding, User attributes), Users, Roles and administrators, Policies (User flows, Identity Experience Framework), and Help. The 'User flows' link under Policies is highlighted with a red box. At the top right, there's a search bar labeled 'Search (Ctrl+ /)' and a button labeled '+ New user flow'. Below the search bar, there are two dropdown menus: 'User flow name' and 'User flow type'. The 'User flow type' dropdown has an option 'Filter by user flow type' with a dropdown arrow. A table below these dropdowns lists entries: Name (Search using user flow name), Type (Mfa), and Mfa.

6. On the Create a user flow page, select the Sign up and sign in user flow.

The screenshot shows the 'Create a user flow' page. At the top, it says 'Home > Azure AD B2C | User flows > Create a user flow'. Below that, a message says 'User flows are predefined, configured policies that you can use to set up authentication experiences for your end users. Select a user flow type to get started.' There's a 'Learn more' link. The next section is titled 'Select a user flow type' and contains six cards:

- Sign up and sign in** (selected): Enables a user to create an account or sign in to their account.
- Profile editing**: Enables a user to configure their user attributes.
- Password reset**: Enables a user to choose a new password after verifying their email.
- Sign up**: Enables a user to create a new account.
- Sign in**: Enables a user to sign in to their account.
- Sign in using resource owner password credentials (ROPC)**: Enables a user with a local account to sign in directly in native applications (no browser required).

7. Under Select a version, select Recommended, and then select Create.



8. Enter a Name for the user flow. For example, signupsignin1.
9. For Identity providers, select Email signup.
10. For User attributes and claims, choose the claims and attributes that you want to collect and send from the user during sign-up. For example, select Show more, and then choose attributes and claims for Country/Region, Display Name, and Postal

Create
X

Create
Sign up and sign in

Info New user flows will now use the OAuth2 implicit grant flow.

Back Select a different type of user flow

Get started with your user flow with a few simple steps:

1. Name
2. Identity providers
3. Multifactor authentication
4. User attributes and claims

1. Name

The unique string used to identify this user flow.

2. Identity providers

Identity providers are the different types of accounts users can log in with.

Please select at least one identity provider.

Email signup

3. Multifactor authentication

Enabling multifactor authentication (MFA) adds an extra layer of security to your user flow.

Multifactor authentication Enabled

4. User attributes and claims

User attributes are values collected on sign up, and claims are values about the user returned to the application in the token. You can create custom attributes for use in your directory.

	Collect attribute	Return claim
City ⓘ	<input type="checkbox"/>	<input type="checkbox"/>
Country/Region ⓘ	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Display Name ⓘ	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Email Address ⓘ	<input type="checkbox"/>	<input type="checkbox"/>
Email Addresses ⓘ	<input type="checkbox"/>	<input type="checkbox"/>
Given Name ⓘ	<input type="checkbox"/>	<input type="checkbox"/>
Identity Provider ⓘ	<input type="checkbox"/>	<input type="checkbox"/>
Identity Provider Access Token ⓘ	<input type="checkbox"/>	<input type="checkbox"/>
Job Title ⓘ	<input type="checkbox"/>	<input type="checkbox"/>
Postal Code ⓘ	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
State/Province ⓘ	<input type="checkbox"/>	<input type="checkbox"/>
Street Address ⓘ	<input type="checkbox"/>	<input type="checkbox"/>
Surname ⓘ	<input type="checkbox"/>	<input type="checkbox"/>
User is new ⓘ	<input type="checkbox"/>	<input type="checkbox"/>
User's Object ID ⓘ	<input type="checkbox"/>	<input type="checkbox"/>

Collect attribute

Given Name ⓘ

Surname ⓘ

City ⓘ

Country/Region ⓘ

Email Address ⓘ

Show more...

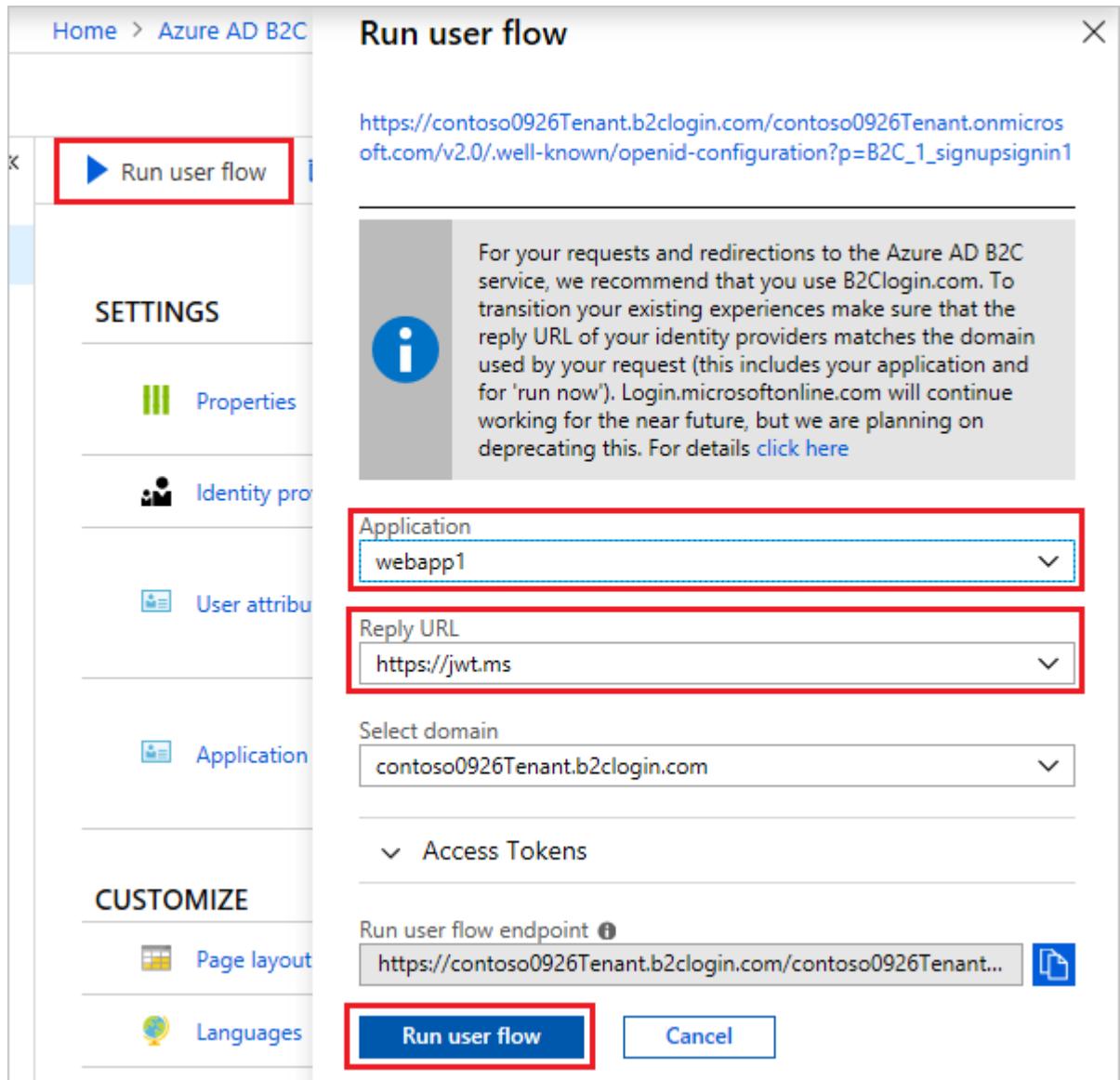
Create
Ok

Code. Click OK.

11. Click Create to add the user flow. A prefix of B2C\_1 is automatically prepended to the name.

## Test the user flow

1. Select the user flow you created to open its overview page, then select Run user flow.
2. For Application, select the web application named webapp1 that you previously registered. The Reply URL should show https://jwt.ms.
3. Click Run user flow, and then select Sign up now.



4. Enter a valid email address, click Send verification code, enter the verification code that you receive, then select Verify code.
5. Enter a new password and confirm the password.
6. Select your country and region, enter the name that you want displayed, enter a postal code, and then click Create. The token is returned to https://jwt.ms and should be displayed to you.

7. You can now run the user flow again and you should be able to sign in with the account that you created. The returned token includes the claims that you selected of country/region, name, and postal code.

## **Enable self-service password reset**

### **To enable self-service password reset for the sign-up or sign-in user flow:**

1. Select the sign-up or sign-in user flow you created.
2. Under Settings in the left menu, select Properties.
3. Under Password complexity, select Self-service password reset.
4. Select Save.

## **Test the user flow**

1. Select the user flow you created to open its overview page, then select Run user flow.
2. For Application, select the web application named webapp1 that you previously registered. The Reply URL should show <https://jwt.ms>.
3. Select Run user flow.
4. From the sign-up or sign-in page, select Forgot your password?.
5. Verify the email address of the account that you previously created, and then select Continue.
6. You now have the opportunity to change the password for the user. Change the password and select Continue. The token is returned to <https://jwt.ms> and should be displayed to you.

## **Create a profile editing user flow**

If you want to enable users to edit their profile in your application, you use a profile editing user flow.

1. In the menu of the Azure AD B2C tenant overview page, select User flows, and then select New user flow.
2. On the Create a user flow page, select the Profile editing user flow.
3. Under Select a version, select Recommended, and then select Create.
4. Enter a Name for the user flow. For example, profileediting1.
5. For Identity providers, select Local Account SignIn.
6. For User attributes, choose the attributes that you want the customer to be able to edit in their profile. For example, select Show more, and then choose both attributes and claims for Display name and Job title. Click OK.
7. Click Create to add the user flow. A prefix of B2C\_1 is automatically appended to the name.

## Test the user flow

1. Select the user flow you created to open its overview page, then select Run user flow.
2. For Application, select the web application named webapp1 that you previously registered. The Reply URL should show <https://jwt.ms>.
3. Click Run user flow, and then sign in with the account that you previously created.
4. You now have the opportunity to change the display name and job title for the user. Click Continue. The token is returned to <https://jwt.ms> and should be displayed to you.

# Azure AD DS

## What is Azure Active Directory Domain Services?

Azure Active Directory Domain Services (AAD DS) is Microsoft's 'managed domain' service in Cloud. It provides a subset of fully compatible traditional AD DS features such as domain join, group policy, DNS service, LDAP, and Kerberos / NTLM authentication.

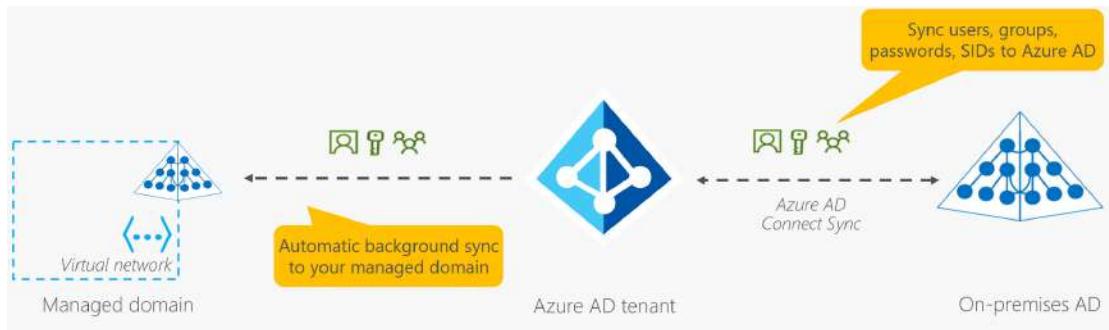
## How does Azure AD DS work?

When you create an Azure AD DS managed domain, you define a unique namespace. This namespace is the domain name, such as aaddscontoso.com. Two Windows Server domain controllers (DCs) are then deployed into your selected Azure region. This deployment of DCs is known as a replica set.

You don't need to manage, configure, or update these DCs. The Azure platform handles the DCs as part of the managed domain, including backups and encryption at rest using Azure Disk Encryption.

A managed domain is configured to perform a one-way synchronization from Azure AD to provide access to a central set of users, groups, and credentials. You can create resources directly in the managed domain, but they aren't synchronized back to Azure AD. Applications, services, and VMs in Azure that connect to the managed domain can then use common AD DS features such as domain join, group policy, LDAP, and Kerberos/NTLM authentication.

In a hybrid environment with an on-premises AD DS environment, Azure AD Connect synchronizes identity information with Azure AD, which is then synchronized to the managed domain.



Azure AD DS replicates identity information from Azure AD, so it works with Azure AD tenants that are cloud-only, or synchronized with an on-premises AD DS environment. The same set of Azure AD DS features exists for both environments.

- If you have an existing on-premises AD DS environment, you can synchronize user account information to provide a consistent identity for users.
- For cloud-only environments, you don't need a traditional on-premises AD DS environment to use the centralized identity services of Azure AD DS.

You can expand a managed domain to have more than one replica set per Azure AD tenant. Replica sets can be added to any peered virtual network in any Azure region that supports Azure AD DS. Additional replica sets in different Azure regions provide geographical disaster recovery for legacy applications if an Azure region goes offline.

## Azure AD DS features and benefits

To provide identity services to applications and VMs in the cloud, Azure AD DS is fully compatible with a traditional AD DS environment for operations such as domain-join, secure LDAP (LDAPS), Group Policy, DNS management, and LDAP bind and read support. LDAP write support is available for objects created in the managed domain, but not resources synchronized from Azure AD.

The following features of Azure AD DS simplify deployment and management operations:

- Simplified deployment experience: Azure AD DS is enabled for your Azure AD tenant using a single wizard in the Azure portal.
- Integrated with Azure AD: User accounts, group memberships, and credentials are automatically available from your Azure AD tenant. New users, groups, or changes to attributes from your Azure AD tenant or your on-premises AD DS environment are automatically synchronized to Azure AD DS.
  - Accounts in external directories linked to your Azure AD aren't available in Azure AD DS. Credentials aren't available for those external directories, so can't be synchronized into a managed domain.
- Use your corporate credentials/passwords: Passwords for users in Azure AD DS are the same as in your Azure AD tenant. Users can use their corporate credentials to

domain-join machines, sign in interactively or over remote desktop, and authenticate against the managed domain.

- NTLM and Kerberos authentication: With support for NTLM and Kerberos authentication, you can deploy applications that rely on Windows-integrated authentication.
- High availability: Azure AD DS includes multiple domain controllers, which provide high availability for your managed domain. This high availability guarantees service uptime and resilience to failures.
  - In regions that support Azure Availability Zones, these domain controllers are also distributed across zones for additional resiliency.
  - Replica sets can also be used to provide geographical disaster recovery for legacy applications if an Azure region goes offline.

Some key aspects of a managed domain include the following:

- The managed domain is a stand-alone domain. It isn't an extension of an on-premises domain.
  - If needed, you can create one-way outbound forest trusts from Azure AD DS to an on-premises AD DS environment.
- Your IT team doesn't need to manage, patch, or monitor domain controllers for this managed domain.

For hybrid environments that run AD DS on-premises, you don't need to manage AD replication to the managed domain. User accounts, group memberships, and credentials from your on-premises directory are synchronized to Azure AD via Azure AD Connect. These user accounts, group memberships, and credentials are automatically available within the managed domain.

## Azure AD (AAD) vs AAD DS

It's important to understand that AAD and AAD DS are two separate services.

- AAD is the directory that sits behind M365/O365 workloads and provides identity and security services. It is not a 'domain service like traditional AD'. It has a flat structure i.e. no Organization Units (OUs).
- AAD DS service is 'domain service' in Cloud and is meant to provide the same service as on-premises AD with domain joins, group policy and has a hierarchical structure with OUs.
- AAD DS requires AAD to be present i.e. an M365 tenant must exist to use AAD DS.

## **‘Joining’ devices to AAD vs AAD DS**

Joining device to AAD provides ‘mobile device management’ (MDM) in combination with Intune whereas AAD DS join is similar to traditional AD domain join.

### **AAD join**

Devices — Windows, Apple (iOS, MAC), Android — can be joined to Azure AD with or without a hybrid deployment that includes an on-premises AD DS environment. AAD join is more internet friendly and allows devices to be joined over the internet and managed using MDM. No domain is required.

It’s suitable for client devices (workstations, mobile devices).

- Personal devices are Azure AD registered,
- Organization owned device not joined to on-premises AD DS are Azure AD joined,
- Organization owned device joined to an on-premises AD DS are Hybrid Azure AD joined.

### **AAD DS domain join**

AAD DS ‘domain join’ is similar to AD domain join and is for corporate owned devices (workstations, servers) that exist within the same network and are managed using Group Policy.

- AAD DS Join is full domain join and requires devices to be within the VNet.
- Suitable for servers that are lifted and shifted or Azure VMs deployed in Azure.
- Microsoft recommendation is to use AAD Join for Windows 10 workstations wherever possible.

### **AAD DS Permissions**

- AAD DS doesn’t provide traditional AD domain or enterprise administrator roles. Hence, no access is available to manage the Domain Controllers (DCs). Instead, admins are added to a built-in ‘AAD DS Administrators’ role that provides all permissions inside the managed domain like ability to create new OUs or group policies.

## Object synchronisation from AAD

AAD DS synchronises objects (Users, Groups) from AAD. Hence, if you have on-premises AD and have configured AAD Connect to sync with Azure AD then objects are synchronised from on-premises all the way to AAD DS.

On-premises AD — — AAD Connect Sync — -> Azure AD — — Automatic Sync — -> AAD DS

- Syncing of objects from AAD to AAD DS is an automatic background process managed by Microsoft.
- When AADDS domain is created, Two OUs are automatically created: “AADDC Users” OU to store all users synced from AAD and “AADDC Computers” OU for all domain joined computers.
- Synchronised objects exist as ‘read-only’ in AAD DS. This means applications that require LDAP write don’t work currently.

## AAD DS domain name and features

AAD DS domain is called a Managed Domain as all infrastructure (DCs, security, high availability etc) is fully taken care by Microsoft.

- As mentioned previously, AAD DS requires an existing M365 tenant.
- VNet: AAD DS domain is exposed inside a VNet. Hence, during creation an existing VNet is required or a wizard lets you create a new VNet. It’s recommended to create the domain within its own subnet separate from other resource subnets for better security.
- Domain name: Any domain name is allowed, even mydomain.local. You can even choose the same domain name as your on-premises domain as long as both domains never see each other i.e. in separate networks.
- One reason to use the same domain name is to lift-and-shift an app that has a dependency on the domain name itself.
- Domain controllers: Service provisions two domain controllers when a domain is created. These DCs are visible in the Azure VNet and are managed by Microsoft itself.
- Group policy: Custom GPO can be created and applied to OU level.
- DNS: Domain provides DNS services for the VNet it’s created in.
- Domain join: Server or workstations within the same VNet can be joined to the domain and managed via GPO.
- On-prem SIDs are synced to SIDHistory in the managed domain. This means existing apps that are lifted and shifted to this domain don’t need any changes.

## AAD DS pricing

Pricing is based on the number of objects which are defined as Users + Groups + domain joined computers.

## AAD DS disaster recovery and availability

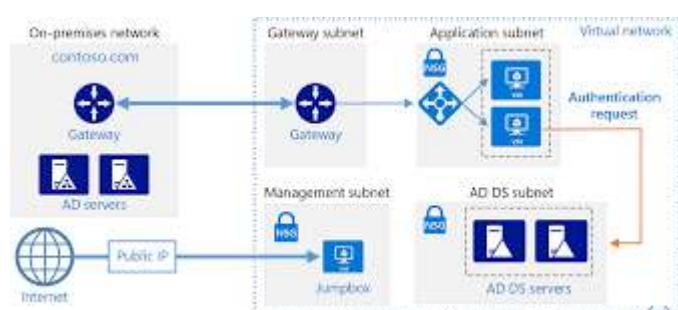
No disaster recovery if VNet goes down. This feature is coming soon.

- Two DCs for high availability with automatic monitoring and incident resolution by Microsoft.

## Deployment scenarios — Why AAD DS and when to use it?

Only use AAD DS if you really need it and have one of the below mentioned scenarios. AAD with MDM (Intune) is the future of the modern workplace and hence, is recommended if your organisation is starting fresh or is purely cloud only with no servers to manage.

- Manage Virtual Machines in the Cloud: If you have lots of VMs deployed in the cloud then AAD DS makes it easier to manage by joining these to a domain and using Group Policy. You can use single organisation level credentials to login to all VMs without worrying about local admin on each VM.
- Applications using Windows Integrated Authentication (WIA) or LDAP bind: Maybe your company doesn't have application source control to convert such applications to modern authentication (OAUTH, Open ID, SAML). Any application that uses WIA and LDAP bind can be lifted-and-shifted to cloud using AAD DS domain.
- ‘Lift-and-shift’ SharePoint server with minor workaround to get user profile sync working.



## Alternatives to AAD DS — Extend on-premises domain to Azure

Create Azure VMs and promote them as replica domain controllers from the on-premises AD DS domain. These domain controllers replicate over a VPN / ExpressRoute connection to the on-premises AD DS environment. The on-premises AD DS domain is effectively extended into Azure.

- Uses VPN/ExpressRoute connections to connect on-prem AD with apps hosted in cloud (Azure),
- Servers are joined directly to on-prem AD via VPN/ExpressRoute or,
- A cloud based domain controller is established and connected to on-prem AD via VPN/ExpressRoute.

## Compare self-managed Active Directory Domain Services, Azure Active Directory, and managed Azure Active Directory Domain Services

To provide applications, services, or devices access to a central identity, there are three common ways to use Active Directory-based services in Azure. This choice in identity solutions gives you the flexibility to use the most appropriate directory for your organization's needs. For example, if you mostly manage cloud-only users that run mobile devices, it may not make sense to build and run your own Active Directory Domain Services (AD DS) identity solution. Instead, you could just use Azure Active Directory.

Although the three Active Directory-based identity solutions share a common name and technology, they're designed to provide services that meet different customer demands. At high level, these identity solutions and feature sets are:

- Active Directory Domain Services (AD DS) - Enterprise-ready lightweight directory access protocol (LDAP) server that provides key features such as identity and authentication, computer object management, group policy, and trusts.
  - AD DS is a central component in many organizations with an on-premises IT environment, and provides core user account authentication and computer management features.
- Azure Active Directory (Azure AD) - Cloud-based identity and mobile device management that provides user account and authentication services for resources such as Microsoft 365, the Azure portal, or SaaS applications.
  - Azure AD can be synchronized with an on-premises AD DS environment to provide a single identity to users that works natively in the cloud.
  - Azure Active Directory Domain Services (Azure AD DS) - Provides managed domain services with a subset of fully-compatible traditional AD DS features such as domain join, group policy, LDAP, and Kerberos / NTLM authentication.

- Azure AD DS integrates with Azure AD, which itself can synchronize with an on-premises AD DS environment. This ability extends central identity use cases to traditional web applications that run in Azure as part of a lift-and-shift strategy.

This overview article compares and contrasts how these identity solutions can work together, or would be used independently, depending on the needs of your organization.

## Azure AD DS and self-managed AD DS

If you have applications and services that need access to traditional authentication mechanisms such as Kerberos or NTLM, there are two ways to provide Active Directory Domain Services in the cloud:

- A managed domain that you create using Azure Active Directory Domain Services (Azure AD DS). Microsoft creates and manages the required resources.
- A self-managed domain that you create and configure using traditional resources such as virtual machines (VMs), Windows Server guest OS, and Active Directory Domain Services (AD DS). You then continue to administer these resources.

With Azure AD DS, the core service components are deployed and maintained for you by Microsoft as a managed domain experience. You don't deploy, manage, patch, and secure the AD DS infrastructure for components like the VMs, Windows Server OS, or domain controllers (DCs).

Azure AD DS provides a smaller subset of features to the traditional self-managed AD DS environment, which reduces some of the design and management complexity. For example, there are no AD forests, domain, sites, and replication links to design and maintain.

For applications and services that run in the cloud and need access to traditional authentication mechanisms such as Kerberos or NTLM, Azure AD DS provides a managed domain experience with the minimal amount of administrative overhead.

When you deploy and run a self-managed AD DS environment, you have to maintain all of the associated infrastructure and directory components. There's additional maintenance overhead with a self-managed AD DS environment, but you're then able to do additional tasks such as extend the schema or create forest trusts.

Common deployment models for a self-managed AD DS environment that provides identity to applications and services in the cloud include the following:

- Standalone cloud-only AD DS - Azure VMs are configured as domain controllers and a separate, cloud-only AD DS environment is created. This AD DS environment doesn't integrate with an on-premises AD DS environment. A different set of credentials is used to sign in and administer VMs in the cloud.
- Resource forest deployment - Azure VMs are configured as domain controllers and an AD DS domain that's part of an existing forest is created. A trust relationship is then

configured to an on-premises AD DS environment. Other Azure VMs can domain-join to this resource forest in the cloud. User authentication runs over a VPN / ExpressRoute connection to the on-premises AD DS environment.

- Extend on-premises domain to Azure - An Azure virtual network connects to an on-premises network using a VPN / ExpressRoute connection. Azure VMs connect to this Azure virtual network, which lets them domain-join to the on-premises AD DS environment.
  - An alternative is to create Azure VMs and promote them as replica domain controllers from the on-premises AD DS domain. These domain controllers replicate over a VPN / ExpressRoute connection to the on-premises AD DS environment. The on-premises AD DS domain is effectively extended into Azure.

The following table outlines some of the features you may need for your organization, and the differences between a managed Azure AD DS domain or a self-managed AD DS domain:

Feature	Azure AD DS	Self-managed AD DS
Managed service	✓	✗
Secure deployments	✓	Administrator secures the deployment
DNS server	✓ (managed service)	✓
Domain or Enterprise administrator privileges	✗	✓
Domain join	✓	✓
Domain authentication using NTLM and Kerberos	✓	✓

Kerberos constrained delegation	Resource-based	Resource-based & account-based
Custom OU structure	✓	✓
Group Policy	✓	✓
Schema extensions	✗	✓
AD domain / forest trusts	✓ (one-way outbound forest trusts only)	✓
Secure LDAP (LDAPS)	✓	✓
LDAP read	✓	✓
LDAP write	✓ (within the managed domain)	✓
Geo-distributed deployments	✓	✓

## Azure AD DS and Azure AD

Azure AD lets you manage the identity of devices used by the organization and control access to corporate resources from those devices. Users can also register their personal device (a bring-your-own (BYO) model) with Azure AD, which provides the device with an identity. Azure AD then authenticates the device when a user signs in to Azure AD and uses the device to access secured resources. The device can be managed using Mobile Device Management

(MDM) software like Microsoft Intune. This management ability lets you restrict access to sensitive resources to managed and policy-compliant devices.

Traditional computers and laptops can also join Azure AD. This mechanism offers the same benefits of registering a personal device with Azure AD, such as to allow users to sign in to the device using their corporate credentials.

Azure AD joined devices give you the following benefits:

- Single-sign-on (SSO) to applications secured by Azure AD.
- Enterprise policy-compliant roaming of user settings across devices.
- Access to the Windows Store for Business using corporate credentials.
- Windows Hello for Business.
- Restricted access to apps and resources from devices compliant with corporate policy.

Devices can be joined to Azure AD with or without a hybrid deployment that includes an on-premises AD DS environment. The following table outlines common device ownership models and how they would typically be joined to a domain:

Type of device	Device platforms	Mechanism
Personal devices	Windows 10, iOS, Android, macOS	Azure AD registered
Organization-owned device not joined to on-premises AD DS	Windows 10	Azure AD joined
Organization-owned device joined to an on-premises AD DS	Windows 10	Hybrid Azure AD joined

On an Azure AD-joined or registered device, user authentication happens using modern OAuth / OpenID Connect based protocols. These protocols are designed to work over the internet, so are great for mobile scenarios where users access corporate resources from anywhere.

With Azure AD DS-joined devices, applications can use the Kerberos and NTLM protocols for authentication, so can support legacy applications migrated to run on Azure VMs as part of a lift-and-shift strategy. The following table outlines differences in how the devices are represented and can authenticate themselves against the directory:

Aspect	Azure AD-joined	Azure AD DS-joined
Device controlled by	Azure AD	Azure AD DS managed domain
Representation in the directory	Device objects in the Azure AD directory	Computer objects in the Azure AD DS managed domain
Authentication	OAuth / OpenID Connect based protocols	Kerberos and NTLM protocols
Management	Mobile Device Management (MDM) software like Intune	Group Policy
Networking	Works over the internet	Must be connected to, or peered with, the virtual network where the managed domain is deployed
Great for...	End-user mobile or desktop devices	Server VMs deployed in Azure

If on-prem AD DS and Azure AD are configured for federated authentication using ADFS then there is no (current/valid) password hash available in Azure DS. Azure AD user accounts created before fed auth was implemented might have an old password hash but this likely doesn't match a hash of their on-prem password. Hence Azure AD DS won't be able to validate the users credentials

# Microsoft Defender for Cloud

## What is Microsoft Defender for Cloud?

Azure Security Center and Azure Defender are now called Microsoft Defender for Cloud. Defender for Cloud is a tool for security posture management and threat protection. It strengthens the security posture of your cloud resources, and with its integrated Microsoft Defender plans, Defender for Cloud protects workloads running in Azure, hybrid, and other cloud platforms.

Defender for Cloud provides the tools needed to harden your resources, track your security posture, protect against cyber attacks, and streamline security management. Because it's natively integrated, deployment of Defender for Cloud is easy, providing you with simple auto provisioning to secure your resources by default.

Defender for Cloud fills three vital needs as you manage the security of your resources and workloads in the cloud and on-premises:



---

Continuous assessment - Understand your current security posture.

Secure score - A single score so that you can tell, at a glance, your current security situation: the higher the score, the lower the identified risk level.

---

---

Secure - Harden all connected resources and services.	Security recommendations - Customized and prioritized hardening tasks to improve your posture. You implement a recommendation by following the detailed remediation steps provided in the recommendation. For many recommendations, Defender for Cloud offers a "Fix" button for automated implementation!
Defend - Detect and resolve threats to those resources and services.	Security alerts - With the enhanced security features enabled, Defender for Cloud detects threats to your resources and workloads. These alerts appear in the Azure portal and Defender for Cloud can also send them by email to the relevant personnel in your organization. Alerts can also be streamed to SIEM, SOAR, or IT Service Management solutions as required.

---

## Posture management and workload protection

Microsoft Defender for Cloud's features cover the two broad pillars of cloud security: cloud security posture management and cloud workload protection.

### Cloud security posture management (CSPM)

In Defender for Cloud, the posture management features provide:

- Visibility - to help you understand your current security situation
- Hardening guidance - to help you efficiently and effectively improve your security

The central feature in Defender for Cloud that enables you to achieve those goals is secure score. Defender for Cloud continually assesses your resources, subscriptions, and organization for security issues. It then aggregates all the findings into a single score so that you can tell, at a glance, your current security situation: the higher the score, the lower the identified risk level.

When you open Defender for Cloud for the first time, it will meet the visibility and strengthening goals as follows:

1. Generate a secure score for your subscriptions based on an assessment of your connected resources compared with the guidance in [Azure Security Benchmark](#). Use the score to understand your security posture, and the compliance dashboard to review your compliance with the built-in benchmark. When you've enabled the enhanced security features, you can customize the standards used to assess your compliance, and add other regulations (such as NIST and Azure CIS) or organization-specific security requirements.
2. Provide hardening recommendations based on any identified security misconfigurations and weaknesses. Use these security recommendations to strengthen the security posture of your organization's Azure, hybrid, and multi-cloud resources.

## Cloud workload protection (CWP)

Defender for Cloud offers security alerts that are powered by Microsoft Threat Intelligence. It also includes a range of advanced, intelligent, protections for your workloads. The workload protections are provided through Microsoft Defender plans specific to the types of resources in your subscriptions. For example, you can enable Microsoft Defender for Storage to get alerted about suspicious activities related to your Azure Storage accounts.

## Azure, hybrid, and multi-cloud protections

Because Defender for Cloud is an Azure-native service, many Azure services are monitored and protected without needing any deployment. When necessary, Defender for Cloud can automatically deploy a Log Analytics agent to gather security-related data. For Azure machines, deployment is handled directly. For hybrid and multi-cloud environments, Microsoft Defender plans are extended to non Azure machines with the help of Azure Arc. CSPM features are extended to multi-cloud machines without the need for any agents.

### Azure-native protections

Defender for Cloud helps you detect threats across:

- Azure PaaS services - Detect threats targeting Azure services including Azure App Service, Azure SQL, Azure Storage Account, and more data services. You can also perform anomaly detection on your Azure activity logs using the native integration with Microsoft Defender for Cloud Apps (formerly known as Microsoft Cloud App Security).
- Azure data services - Defender for Cloud includes capabilities that help you automatically classify your data in Azure SQL. You can also get assessments for potential vulnerabilities across Azure SQL and Storage services, and recommendations for how to mitigate them.
- Networks - Defender for Cloud helps you limit exposure to brute force attacks. By reducing access to virtual machine ports, using the just-in-time VM access, you can harden your network by preventing unnecessary access. You can set secure access policies on selected ports, for only authorized users, allowed source IP address ranges or IP addresses, and for a limited amount of time.

### Defend your hybrid resources

In addition to defending your Azure environment, you can add Defender for Cloud capabilities to your hybrid cloud environment to protect your non-Azure servers. To help you focus on what matters the most, you'll get customized threat intelligence and prioritized alerts according to your specific environment.

To extend protection to on-premises machines, deploy Azure Arc and enable Defender for Cloud's enhanced security features. Learn more in Add non-Azure machines with Azure Arc.

## Defend resources running on other clouds

Defender for Cloud can protect resources in other clouds (such as AWS and GCP).

For example, if you've connected an Amazon Web Services (AWS) account to an Azure subscription, you can enable any of these protections:

- Defender for Cloud's CSPM features extend to your AWS resources. This agentless plan assesses your AWS resources according to AWS-specific security recommendations and these are included in your secure score. The resources will also be assessed for compliance with built-in standards specific to AWS (AWS CIS, AWS PCI DSS, and AWS Foundational Security Best Practices). Defender for Cloud's asset inventory page is a multi-cloud enabled feature helping you manage your AWS resources alongside your Azure resources.
- Microsoft Defender for Kubernetes extends its container threat detection and advanced defenses to your Amazon EKS Linux clusters.
- Microsoft Defender for servers brings threat detection and advanced defenses to your Windows and Linux EC2 instances. This plan includes the integrated license for Microsoft Defender for Endpoint, security baselines and OS level assessments, vulnerability assessment scanning, adaptive application controls (AAC), file integrity monitoring (FIM), and more.

## Vulnerability assessment and management

Continuously Assess <small>(Know your security posture. Identify and track vulnerabilities.)</small>	Secure <small>(Harden resources and services with Azure Security Benchmark)</small>	Defend <small>(Detect and resolve threats to resources and services)</small>
<ul style="list-style-type: none"><li>• Secure score</li><li>• Vulnerability assessments</li><li>• Asset inventory</li><li>• Regulatory compliance</li><li>• File integrity monitoring</li></ul>	<ul style="list-style-type: none"><li>• Security recommendations</li><li>• Just-in-time VM access</li><li>• Adaptive network hardening</li><li>• Adaptive application control</li></ul>	<ul style="list-style-type: none"><li>• Microsoft Defender</li><li>• Security alerts</li><li>• Integration with Microsoft Sentinel (or other SIEM)</li></ul>

Defender for Cloud includes vulnerability assessment solutions for your virtual machines, container registries, and SQL servers as part of the enhanced security features. Some of the scanners are powered by Qualys. But you don't need a Qualys license, or even a Qualys account - everything's handled seamlessly inside Defender for Cloud.

Microsoft Defender for servers includes automatic, native integration with Microsoft Defender for Endpoint.

## Optimize and improve security by configuring recommended controls

Continuously Assess  (Know your security posture. Identify and track vulnerabilities.)	Secure  (Harden resources and services with Azure Security Benchmark)	Defend  (Detect and resolve threats to resources and services)
<ul style="list-style-type: none"><li>• Secure score</li><li>• Vulnerability assessments</li><li>• Asset inventory</li><li>• Regulatory compliance</li><li>• File integrity monitoring</li></ul>	<ul style="list-style-type: none"><li>• Security recommendations</li><li>• Just-in-time VM access</li><li>• Adaptive network hardening</li><li>• Adaptive application control</li></ul>	<ul style="list-style-type: none"><li>• Microsoft Defender</li><li>• Security alerts</li><li>• Integration with Microsoft Sentinel (or other SIEM)</li></ul>

It's a security basics to know and make sure your workloads are secure, and it starts with having tailored security policies in place. Because policies in Defender for Cloud are built on top of Azure Policy controls, you're getting the full range and flexibility of a world-class policy solution. In Defender for Cloud, you can set your policies to run on management groups, across subscriptions, and even for a whole tenant.

Defender for Cloud continuously discovers new resources that are being deployed across your workloads and assesses whether they are configured according to security best practices. If not, they're flagged and you get a prioritized list of recommendations for what you need to fix. Recommendations help you reduce the attack surface across each of your resources.

The list of recommendations is enabled and supported by the Azure Security Benchmark. This Microsoft-authored, Azure-specific, benchmark provides a set of guidelines for security and compliance best practices based on common compliance frameworks.

In this way, Defender for Cloud enables you not just to set security policies, but to apply secure configuration standards across your resources.

## Management ports of virtual machines should be protected with just-in-time network access control

...



Exempt View policy definition Open query

Severity

High

Freshness interval

24 Hours

Exempted resources



44

[View all exemptions](#)

### Description

Defender for Cloud has identified some overly-permissive inbound rules for management ports in your network security group. Enable just-in-time access control to protect your machine from internet-based brute-force attacks. [Learn more.](#)

### Remediation steps

### Affected resources

[Unhealthy resources \(15\)](#) [Healthy resources \(93\)](#) [Not applicable resources \(76\)](#)

Search virtual machines

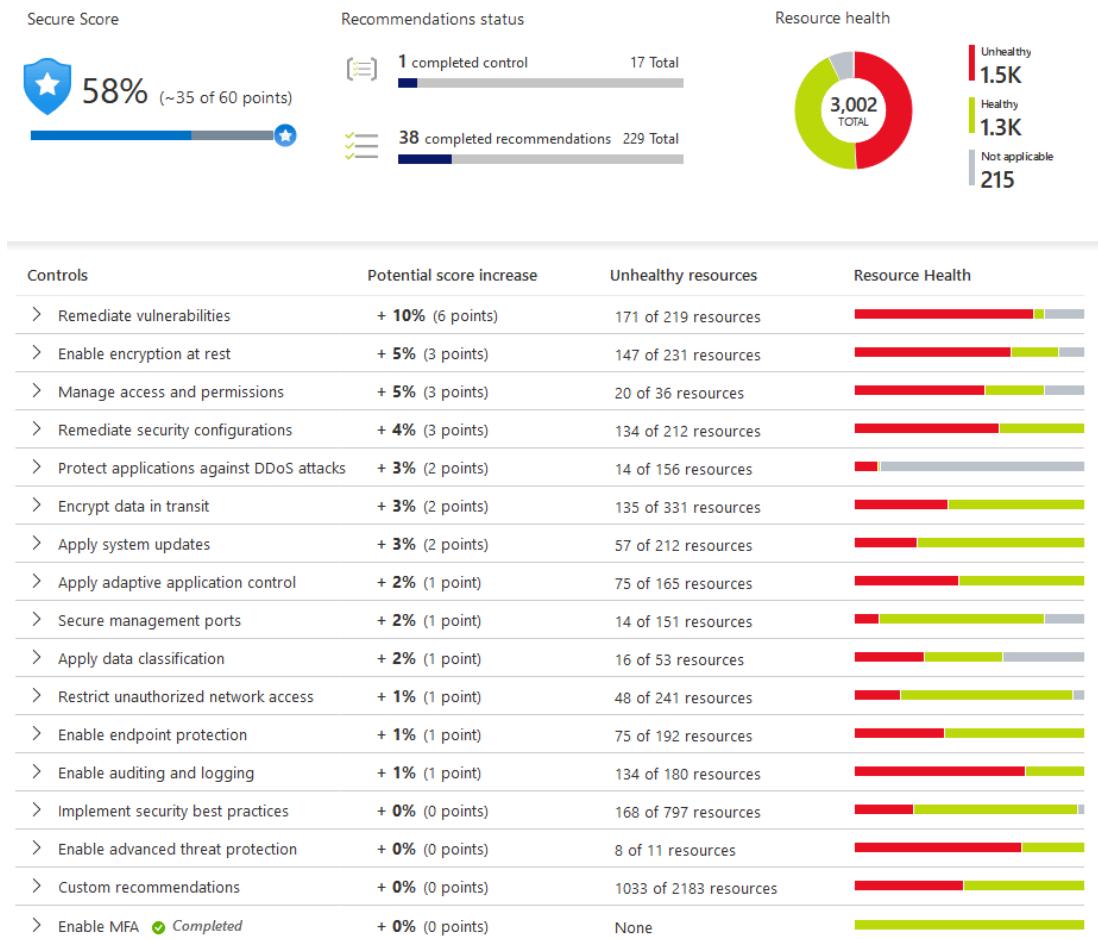
Name ↑↓ Subscription

[sqlonvm](#) Rome Core Utils

[shir-sap](#) CyberSecSOC

[shir-hive](#) CyberSecSOC

To help you understand how important each recommendation is to your overall security posture, Defender for Cloud groups the recommendations into security controls and adds a secure score value to each control. This is crucial in enabling you to prioritize your security work.



## Defend against threats

Continuously Assess	Secure	Defend
(Know your security posture. Identify and track vulnerabilities.)	(Harden resources and services with Azure Security Benchmark)	(Detect and resolve threats to resources and services)
<ul style="list-style-type: none"> <li>Secure score</li> <li>Vulnerability assessments</li> <li>Asset inventory</li> <li>Regulatory compliance</li> <li>File integrity monitoring</li> </ul>	<ul style="list-style-type: none"> <li>Security recommendations</li> <li>Just-in-time VM access</li> <li>Adaptive network hardening</li> <li>Adaptive application control</li> </ul>	<ul style="list-style-type: none"> <li>Microsoft Defender</li> <li>Security alerts</li> <li>Integration with Microsoft Sentinel (or other SIEM)</li> </ul>

Defender for Cloud provides:

- Security alerts - When Defender for Cloud detects a threat in any area of your environment, it generates a security alert. These alerts describe details of the affected resources, suggested remediation steps, and in some cases an option to trigger a logic app in response. Whether an alert is generated by Defender for Cloud, or received by Defender for Cloud from an integrated security product, you can export it. To export your alerts to Microsoft Sentinel, any third-party SIEM, or any other external tool, Defender for Cloud's threat protection includes fusion kill-chain analysis, which automatically correlates alerts in your environment based on cyber kill-chain analysis,

to help you better understand the full story of an attack campaign, where it started and what kind of impact it had on your resources.

- Advanced threat protection features for virtual machines, SQL databases, containers, web applications, your network, and more - Protections include securing the management ports of your VMs with just-in-time access, and adaptive application controls to create allowlists for what apps should and shouldn't run on your machines.

## Microsoft Defender for Cloud's enhanced security features

The enhanced security features are free for the first 30 days. At the end of 30 days, if you decide to continue using the service, it'll automatically start charging for usage.

The screenshot shows the 'Settings | Defender plans' section for 'Contoso Infra2'. It compares two states:

Enhanced security off	Enable all Microsoft Defender for Cloud plans
✓ Continuous assessment and security recommendations	✓ Continuous assessment and security recommendations
✓ Secure score	✓ Secure score
✗ Just in time VM Access	✓ Just in time VM Access
✗ Adaptive application controls and network hardening	✓ Adaptive application controls and network hardening
✗ Regulatory compliance dashboard and reports	✓ Regulatory compliance dashboard and reports
✗ Threat protection for Azure VMs and non-Azure servers (including Server EDR)	✓ Threat protection for Azure VMs and non-Azure servers (including Server EDR)
✗ Threat protection for supported PaaS services	✓ Threat protection for supported PaaS services

## What are the benefits of enabling enhanced security features?

Defender for Cloud is offered in two modes:

- Without enhanced security features (Free) - Defender for Cloud is enabled for free on all your Azure subscriptions when you visit the workload protection dashboard in the Azure portal for the first time, or if enabled programmatically via API. Using this free mode provides the secure score and its related features: security policy, continuous security assessment, and actionable security recommendations to help you protect your Azure resources.
- Defender for Cloud with all enhanced security features - Enabling enhanced security extends the capabilities of the free mode to workloads running in private and other public clouds, providing unified security management and threat protection across your hybrid cloud workloads. Some of the major benefits include:
  - Microsoft Defender for Endpoint - Microsoft Defender for servers includes Microsoft Defender for Endpoint for comprehensive endpoint detection and response (EDR).

- Vulnerability assessment for virtual machines, container registries, and SQL resources - Easily enable vulnerability assessment solutions to discover, manage, and resolve vulnerabilities. View, investigate, and remediate the findings directly from within Defender for Cloud.
- Multi-cloud security - Connect your accounts from Amazon Web Services (AWS) and Google Cloud Platform (GCP) to protect resources and workloads on those platforms with a range of Microsoft Defender for Cloud security features.
- Hybrid security – Get a unified view of security across all of your on-premises and cloud workloads. Apply security policies and continuously assess the security of your hybrid cloud workloads to ensure compliance with security standards. Collect, search, and analyze security data from multiple sources, including firewalls and other partner solutions.
- Threat protection alerts - Advanced behavioral analytics and the Microsoft Intelligent Security Graph provide an edge over evolving cyber-attacks. Built-in behavioral analytics and machine learning can identify attacks and zero-day exploits. Monitor networks, machines, data stores (SQL servers hosted inside and outside Azure, Azure SQL databases, Azure SQL Managed Instance, and Azure Storage) and cloud services for incoming attacks and post-breach activity. Streamline investigation with interactive tools and contextual threat intelligence.
- Track compliance with a range of standards - Defender for Cloud continuously assesses your hybrid cloud environment to analyze the risk factors according to the controls and best practices in Azure Security Benchmark. When you enable the enhanced security features, you can apply a range of other industry standards, regulatory standards, and benchmarks according to your organization's needs. Add standards and track your compliance with them from the regulatory compliance dashboard.
- Access and application controls - Block malware and other unwanted applications by applying machine learning powered recommendations adapted to your specific workloads to create allow and blocklists. Reduce the network attack surface with just-in-time, controlled access to management ports on Azure VMs. Access and application controls drastically reduce exposure to brute force and other network attacks.
- Container security features - Benefit from vulnerability management and real-time threat protection on your containerized environments. Charges are based on the number of unique container images pushed to your connected registry. After an image has been scanned once, you won't be charged for it again unless it's modified and pushed once more.
- Breadth threat protection for resources connected to Azure - Cloud-native threat protection for the Azure services common to all of your resources: Azure Resource Manager, Azure DNS, Azure network layer, and Azure Key Vault. Defender for Cloud has unique visibility into the Azure management

layer and the Azure DNS layer, and can therefore protect cloud resources that are connected to those layers.

# Azure Key Vault

## About Azure Key Vault

Azure Key Vault helps solve the following problems:

- Secrets Management - Azure Key Vault can be used to Securely store and tightly control access to tokens, passwords, certificates, API keys, and other secrets
- Key Management - Azure Key Vault can be used as a Key Management solution. Azure Key Vault makes it easy to create and control the encryption keys used to encrypt your data.
- Certificate Management - Azure Key Vault lets you easily provision, manage, and deploy public and private Transport Layer Security/Secure Sockets Layer (TLS/SSL) certificates for use with Azure and your internal connected resources.

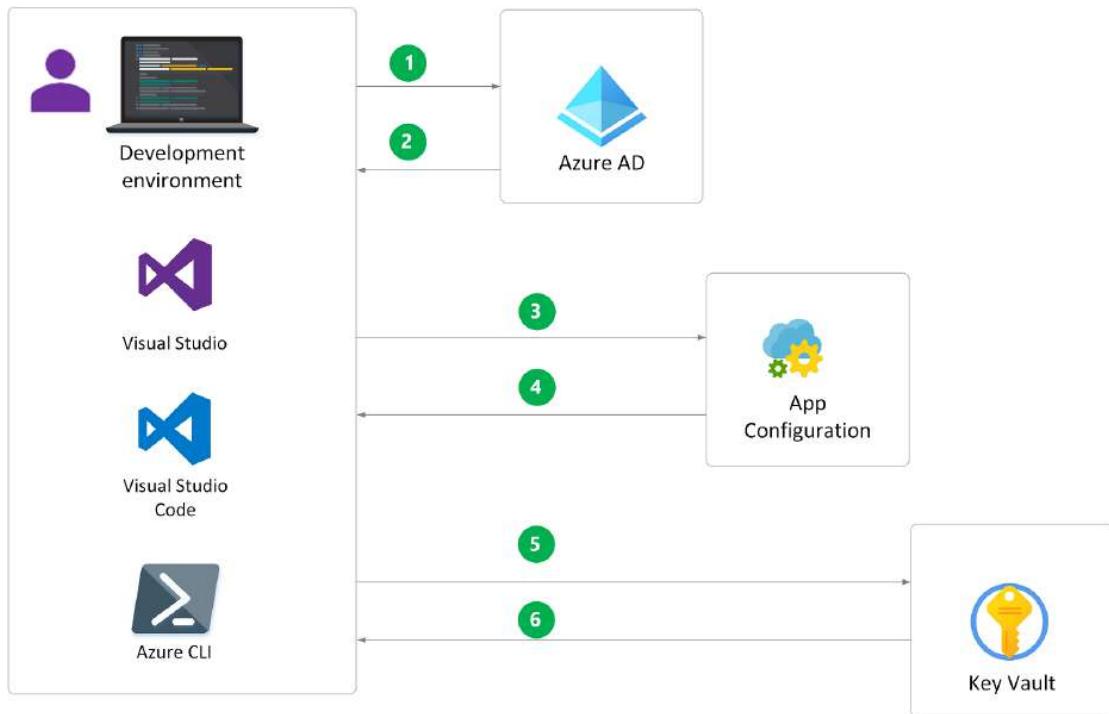


Azure Key Vault has two service tiers: Standard, which encrypts with a software key, and a Premium tier, which includes hardware security module(HSM)-protected keys.

## Why use Azure Key Vault?

### Centralize application secrets

Centralizing storage of application secrets in Azure Key Vault allows you to control their distribution. Key Vault greatly reduces the chances that secrets may be accidentally leaked. When using Key Vault, application developers no longer need to store security information in their application. Not having to store security information in applications eliminates the need to make this information part of the code. For example, an application may need to connect to a database. Instead of storing the connection string in the app's code, you can store it securely in Key Vault.



Your applications can securely access the information they need by using URIs. These URIs allow the applications to retrieve specific versions of a secret. There is no need to write custom code to protect any of the secret information stored in Key Vault.

### Securely store secrets and keys

Access to a key vault requires proper authentication and authorization before a caller (user or application) can get access. Authentication establishes the identity of the caller, while authorization determines the operations that they are allowed to perform.

Authentication is done via Azure Active Directory. Authorization may be done via Azure role-based access control (Azure RBAC) or Key Vault access policy. Azure RBAC can be used for both management of the vaults and access data stored in a vault, while key vault access policy can only be used when attempting to access data stored in a vault.

Azure Key Vaults may be either software-protected or, with the Azure Key Vault Premium tier, hardware-protected by hardware security modules (HSMs). Software-protected keys, secrets, and certificates are safeguarded by Azure, using industry-standard algorithms and key lengths. For situations where you require added assurance, you can import or generate keys in HSMs that never leave the HSM boundary. Azure Key Vault uses nCipher HSMs, which are Federal Information Processing Standards (FIPS) 140-2 Level 2 validated. You can use nCipher tools to move a key from your HSM to Azure Key Vault.

Finally, Azure Key Vault is designed so that Microsoft does not see or extract your data.

## Retrieve a secret from Key Vault

Home > AKV-Contoso > AKV-Contoso > testsecret >

 **bbb31809b4bd4997a6accaa4c85f45db** ⚡ ...  
Secret Version

 Save  Discard changes

### Properties

Created 10/21/2021, 11:02:15 AM

Updated 10/21/2021, 11:02:15 AM

Secret Identifier [https://akv-contoso.vault.azure....](https://akv-contoso.vault.azure.net/secrets/bbb31809b4bd4997a6accaa4c85f45db) 

### Settings

Set activation date 

Set expiration date 

Enabled  Yes 

Tags 0 tags

### Secret

Content type (optional)

**Show Secret Value**

Secret value

.....



By clicking the "Show Secret Value" button in the right pane, you can see the hidden value.

Secret

Content type (optional)

**Hide Secret Value**

Secret value

Kw4IP^%pAi70



## Monitor access and use

Once you have created a couple of Key Vaults, you will want to monitor how and when your keys and secrets are being accessed. You can monitor activity by enabling logging for your vaults. You can configure Azure Key Vault to:

- Archive to a storage account.
- Stream to an event hub.
- Send the logs to Azure Monitor logs.

You have control over your logs and you may secure them by restricting access and you may also delete logs that you no longer need.

## Simplified administration of application secrets

When storing valuable data, you must take several steps. Security information must be secured, it must follow a life cycle, and it must be highly available. Azure Key Vault simplifies the process of meeting these requirements by:

- Removing the need for in-house knowledge of Hardware Security Modules.
- Scaling up on short notice to meet your organization's usage spikes.
- Replicating the contents of your Key Vault within a region and to a secondary region. Data replication ensures high availability and takes away the need of any action from the administrator to trigger the failover.
- Providing standard Azure administration options via the portal, Azure CLI and PowerShell.
- Automating certain tasks on certificates that you purchase from Public CAs, such as enrollment and renewal.

In addition, Azure Key Vaults allow you to segregate application secrets. Applications may access only the vault that they are allowed to access, and they can be limited to only perform specific operations. You can create an Azure Key Vault per application and restrict the secrets stored in a Key Vault to a specific application and team of developers.

## Azure Key Vault keys, secrets and certificates

Azure Key Vault enables Microsoft Azure applications and users to store and use several types of secret/key data. Key Vault resource provider supports two resource types: vaults and managed HSMs.

### DNS suffixes for base URL

The table below shows the base URL DNS suffix used by the data-plane endpoint for vaults and managed HSM pools in various cloud environments.

Cloud environment	DNS suffix for vaults	DNS suffix for managed HSMs
Azure Cloud	.vault.azure.net	.managedhsm.azure.net
Azure China Cloud	.vault.azure.cn	Not supported
Azure US Government	.vault.usgovcloudapi.net	Not supported
Azure German Cloud	.vault.microsoftazure.de	Not supported

### Object types

The table below shows object types and their suffixes in the base URL.

Object type	URL Suffix	Vaults	Managed HSM Pools

HSM-protected keys	/keys	Supported	Supported
Software-protected keys	/keys	Supported	Not supported
Secrets	/secrets	Supported	Not supported
Certificates	/certificates	Supported	Not supported
Storage account keys	/storage	Supported	Not supported

- Cryptographic keys: Supports multiple key types and algorithms, and enables the use of software-protected and HSM-protected keys.
- Secrets: Provides secure storage of secrets, such as passwords and database connection strings.
- Certificates: Supports certificates, which are built on top of keys and secrets and add an automated renewal feature. Keep in mind when a certificate is created, an addressable key and secret are also created with the same name.
- Azure Storage account keys: Can manage keys of an Azure Storage account for you. Internally, Key Vault can list (sync) keys with an Azure Storage Account, and regenerate (rotate) the keys periodically.

## Data types

Refer to the JOSE specifications for relevant data types for keys, encryption, and signing.

- algorithm - a supported algorithm for a key operation, for example, RSA1\_5
- ciphertext-value - cipher text octets, encoded using Base64URL
- digest-value - the output of a hash algorithm, encoded using Base64URL
- key-type - one of the supported key types, for example RSA (Rivest-Shamir-Adleman).

- plaintext-value - plaintext octets, encoded using Base64URL
- signature-value - output of a signature algorithm, encoded using Base64URL
- base64URL - a Base64URL [RFC4648] encoded binary value
- boolean - either true or false
- Identity - an identity from Azure Active Directory (AAD).
- IntDate - a JSON decimal value representing the number of seconds from 1970-01-01T0:0:0Z UTC until the specified UTC date/time. See RFC3339 for details regarding date/times, in general and UTC in particular.

## Objects, identifiers, and versioning

Objects stored in Key Vault are versioned whenever a new instance of an object is created. Each version is assigned a unique identifier and URL. When an object is first created, it's given a unique version identifier and marked as the current version of the object. Creation of a new instance with the same object name gives the new object a unique version identifier, causing it to become the current version.

Objects in Key Vault can be addressed by specifying a version or by omitting version for operations on the current version of the object. For example, given a Key with the name MasterKey, performing operations without specifying a version causes the system to use the latest available version. Performing operations with the version-specific identifier causes the system to use that specific version of the object.

## Vault-name and Object-name

Objects are uniquely identified within Key Vault using a URL. No two objects in the system have the same URL, regardless of geo-location. The complete URL to an object is called the Object Identifier. The URL consists of a prefix that identifies the Key Vault, object type, user provided Object Name, and an Object Version. The Object Name is case-insensitive and immutable. Identifiers that don't include the Object Version are referred to as Base Identifiers.

An object identifier has the following general format (depending on container type):

- For Vaults: `https://{{vault-name}}.vault.azure.net/{{object-type}}/{{object-name}}/{{object-version}}`
- For Managed HSM pools: `https://{{hsm-name}}.managedhsm.azure.net/{{object-type}}/{{object-name}}/{{object-version}}`

Element	Description
vault-name or hsm-name	<p>The name for a vault or an Managed HSM pool in the Microsoft Azure Key Vault service.</p> <p>Vault names and Managed HSM pool names are selected by the user and are globally unique.</p> <p>Vault name and Managed HSM pool name must be a 3-24 character string, containing only 0-9, a-z, A-Z, and -.</p>
object-type	The type of the object, "keys", "secrets", or 'certificates'.
object-name	An object-name is a user provided name for and must be unique within a Key Vault. The name must be a 1-127 character string, starting with a letter and containing only 0-9, a-z, A-Z, and -.
object-version	An object-version is a system-generated, 32 character string identifier that is optionally used to address a unique version of an object.

## Best practices for using Azure Key Vault

Azure Key Vault safeguards encryption keys and secrets like certificates, connection strings, and passwords.

### Use separate key vaults

Our recommendation is to use a vault per application per environment (development, pre-production, and production), per region. This helps you not share secrets across environments and regions. It will also reduce the threat in case of a breach.

### Why separate key vaults are recommended

Key vaults define security boundaries for stored secrets. Grouping secrets into the same vault increases the blast radius of a security event because attacks might be able to access secrets across concerns. To mitigate access across concerns, consider what secrets a

specific application should have access to, and then separate your key vaults based on this delineation. Separating key vaults by application is the most common boundary. Security boundaries, however, can be more granular for large applications, for example, per group of related services.

## Control access to your vault

Encryption keys and secrets like certificates, connection strings, and passwords are sensitive and business critical. You need to secure access to your key vaults by allowing only authorized applications and users.

Suggestions for controlling access to your vault are as follows:

- Lock down access to your subscription, resource group, and key vaults (role-based access control (RBAC)).
- Create access policies for every vault.
- Use the principle of least privilege access to grant access.
- Turn on firewall and virtual network service endpoints.

## Backup

Make sure you take regular backups of your vault. Backups should be performed when you update, delete, or create objects in your vault.

## Turn on logging

Turn on logging for your vault. Also, set up alerts.

## Turn on recovery options

- Turn on soft-delete.
- Turn on purge protection if you want to guard against force deletion of the secrets and key vault even after soft-delete is turned on.

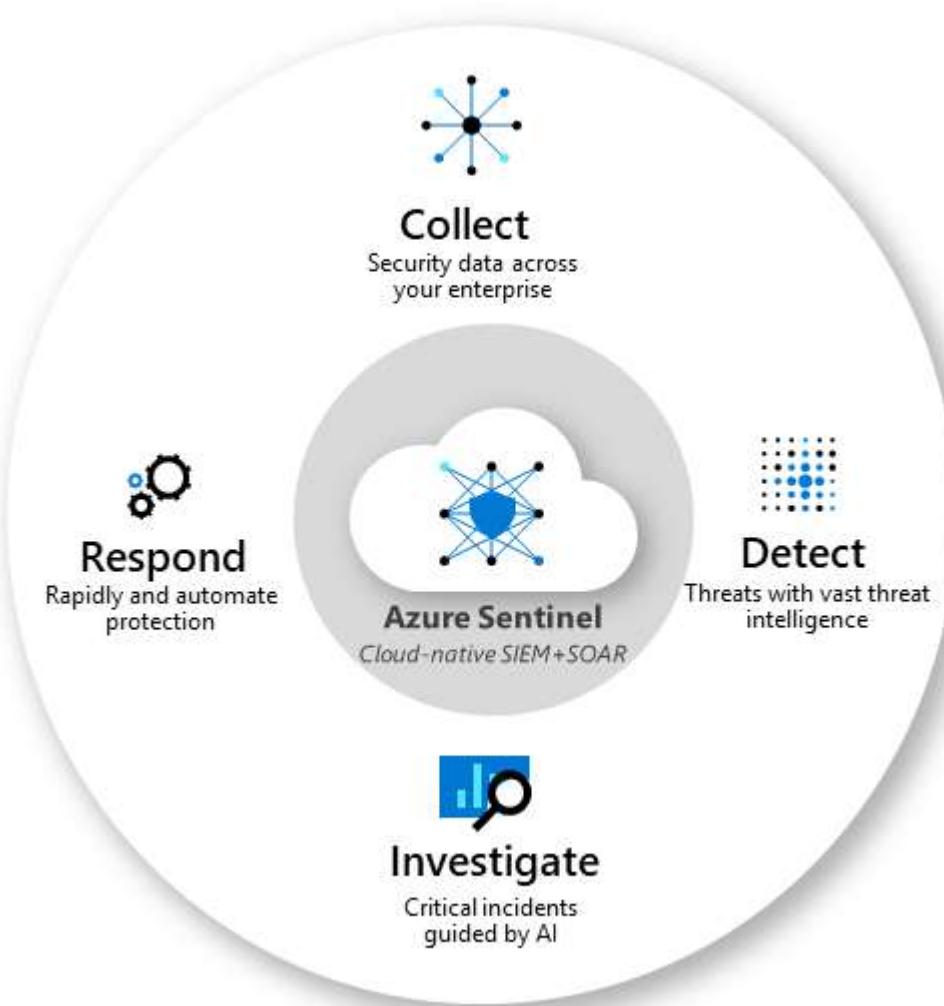
# Azure Sentinel

## What is Microsoft Sentinel?

Azure Sentinel is now called Microsoft Sentinel. Microsoft Sentinel is a scalable, cloud-native, security information and event management (SIEM) and security orchestration, automation, and response (SOAR) solution. Microsoft Sentinel delivers intelligent security analytics and threat intelligence across the enterprise, providing a single solution for attack detection, threat visibility, proactive hunting, and threat response.

Microsoft Sentinel is your birds-eye view across the enterprise alleviating the stress of increasingly sophisticated attacks, increasing volumes of alerts, and long resolution time frames.

- Collect data at cloud scale across all users, devices, applications, and infrastructure, both on-premises and in multiple clouds.
- Detect previously undetected threats, and minimize false positives using Microsoft's analytics and unparalleled threat intelligence.
- Investigate threats with artificial intelligence, and hunt for suspicious activities at scale, tapping into years of cyber security work at Microsoft.
- Respond to incidents rapidly with built-in orchestration and automation of common tasks.



Building on the full range of existing Azure services, Microsoft Sentinel natively incorporates proven foundations, like Log Analytics, and Logic Apps. Microsoft Sentinel enriches your investigation and detection with AI, and provides Microsoft's threat intelligence stream and enables you to bring your own threat intelligence.

## Connect to all your data

To on-board Microsoft Sentinel, you first need to connect to your security sources.

Microsoft Sentinel comes with a number of connectors for Microsoft solutions, available out of the box and providing real-time integration, including Microsoft 365 Defender (formerly Microsoft Threat Protection) solutions, and Microsoft 365 sources, including Office 365, Azure AD, Microsoft Defender for Identity (formerly Azure ATP), and Microsoft Defender for Cloud Apps, and more. In addition, there are built-in connectors to the broader security ecosystem for non-Microsoft solutions. You can also use common event format, Syslog or REST-API to connect your data sources with Microsoft Sentinel as well.

The screenshot shows the Azure Sentinel Data connectors page. On the left, a sidebar lists General, Threat management, Configuration, and Data connectors (which is selected). The main area displays a summary of 97 connectors, 15 connected, and 0 coming soon. A search bar and filters for Providers, Data Types, and Status are present. Below this, a list of connectors includes Agari Phishing Defense and Brand Protection (Preview), AI Analyst Darktrace (Preview), AI Vectra Detect (Preview), Akamai Security Events (Preview), Alcide kAudit (Preview), Alsid for Active Directory (Preview), Amazon Web Services, and Apache HTTP Server (Preview). On the right, a detailed view of the AI Vectra Detect (Preview) connector is shown, including its status as 'Not connected', provider as 'Vectra AI', and last log received. It has a description about connecting Vectra Detect logs to Azure Sentinel for improved investigation. Related content links to 1 Workbook, 4 Queries, and 0 Analytic rules templates, with a 'Open connector page' button at the bottom.

## Workbooks

After you connect your data sources to Microsoft Sentinel, you can monitor the data using the Microsoft Sentinel integration with Azure Monitor Workbooks, which provides versatility in creating custom workbooks.

The screenshot shows the Azure Sentinel Workbooks interface. On the left, there's a navigation sidebar with categories like General, Threat management, Configuration, and Threat intelligence (Preview). The 'Workbooks' section is selected. In the center, a list of workbooks is displayed, including 'AI Analyst Darktrace Model Breach Summary', 'AI Vectra Detect', 'AISID for AD | Indicators of Exposure', 'Analytics Efficiency', 'ASC Compliance and Protection', 'AWS Network Activities', 'AWS User Activities', and 'Azure Activity'. A specific workbook, 'Analytics Efficiency' by MICROSOFT, is highlighted and shown in a preview pane on the right. The preview pane includes a description: 'Gain insights into the efficacy of your analytics rules. In the workbook you can analyze and monitor the analytics rules found in your workspace to achieve better performance by your SOC.' It also lists 'Required data types:' with three items: 'SecurityAlert' and 'SecurityIncident' (both checked) and 'Event'.

- Workbooks are intended for SOC engineers and analysts of all tiers to visualize data.
- While Workbooks are best used for high-level views of Microsoft Sentinel data, and require no coding knowledge, you cannot integrate Workbooks with external data.

## Analytics

To help you reduce noise and minimize the number of alerts you have to review and investigate, Microsoft Sentinel uses analytics to correlate alerts into incidents. Incidents are groups of related alerts that together create an actionable possible-threat that you can investigate and resolve. Use the built-in correlation rules as-is, or use them as a starting point to build your own. Microsoft Sentinel also provides machine learning rules to map your network behavior and then look for anomalies across your resources. These analytics connect the dots, by combining low fidelity alerts about different entities into potential high-fidelity security incidents.

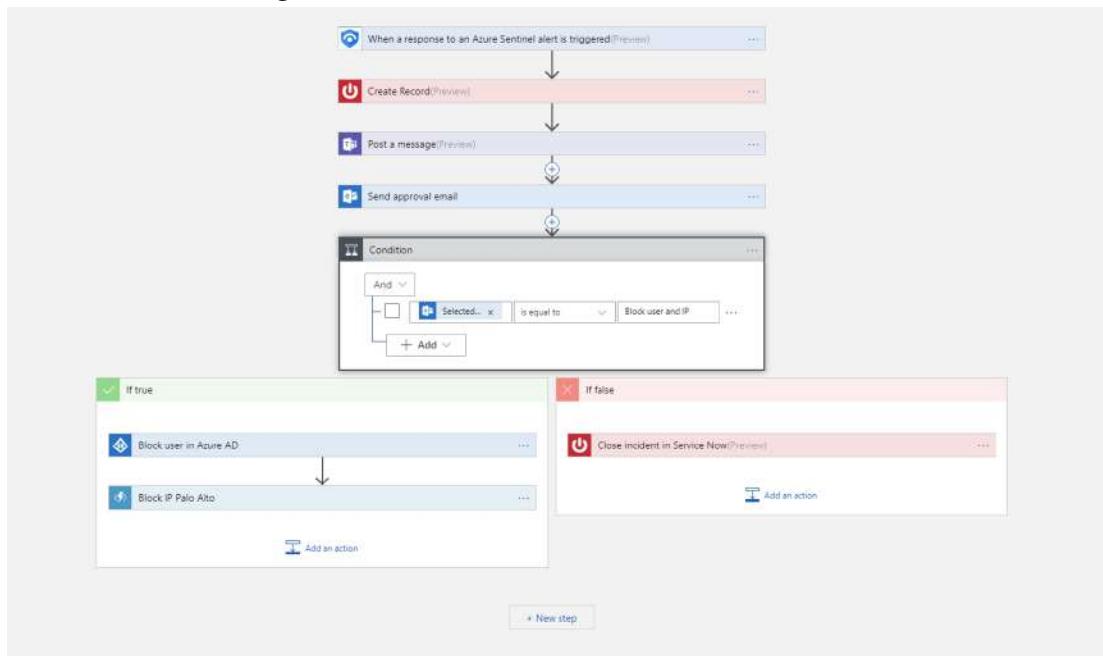
The screenshot shows the Azure Sentinel Incidents interface. On the left, there's a navigation sidebar with categories like General, Threat management, Configuration, and Threat intelligence (Preview). The 'Incidents' section is selected. In the center, a list of incidents is displayed, showing '178 Open incidents' and '178 New incidents'. A detailed view of an incident titled 'ADFS DKM Master Key Export' is shown on the right. The incident details include the owner ('Unassigned'), status ('New'), and severity ('High'). The description states: 'Identifies an exfiltration attempt of the ADFS DKM Master Key from Active Directory. References: - https://www.microsoft.com/the-microsoft-attack-vector-project-reports-a-potential-exploit-attack-leverages-solar-winds-supply-chain-compromises-with-sunburst... Show more ». Alert product names: - Azure Sentinel'. Below the incident details, there are sections for Evidence (132 Events, 1 Alerts, 0 Bookmarks), Last update time (05/03/21, 12:14 PM), and Creation time (05/03/21, 12:14 PM).

## Security automation & orchestration

Automate your common tasks and simplify security orchestration with playbooks that integrate with Azure services and your existing tools.

Built on the foundation of Azure Logic Apps, Microsoft Sentinel's automation and orchestration solution provides a highly extensible architecture that enables scalable automation as new technologies and threats emerge. To build playbooks with Azure Logic Apps, you can choose from a growing gallery of built-in playbooks. These include 200+ connectors for services such as Azure functions. The connectors allow you to apply any custom logic in code, ServiceNow, Jira, Zendesk, HTTP requests, Microsoft Teams, Slack, Windows Defender ATP, and Defender for Cloud Apps.

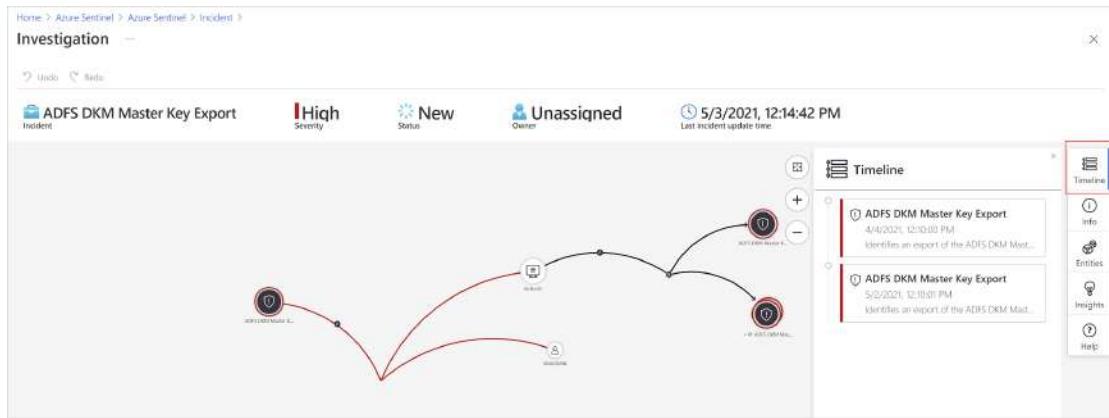
For example, if you use the ServiceNow ticketing system, you can use the tools provided to use Azure Logic Apps to automate your workflows and open a ticket in ServiceNow each time a particular event is detected.



- Playbooks are intended for SOC engineers and analysts of all tiers, to automate and simplify tasks, including data ingestion, enrichment, investigation, and remediation.
- Playbooks work best with single, repeatable tasks, and require no coding knowledge. Playbooks are not suitable for ad-hoc or complex task chains, or for documenting and sharing evidence.

## Investigation

Currently in preview, Microsoft Sentinel deep investigation tools help you to understand the scope and find the root cause of a potential security threat. You can choose an entity on the interactive graph to ask interesting questions for a specific entity, and drill down into that entity and its connections to get to the root cause of the threat.



## Hunting

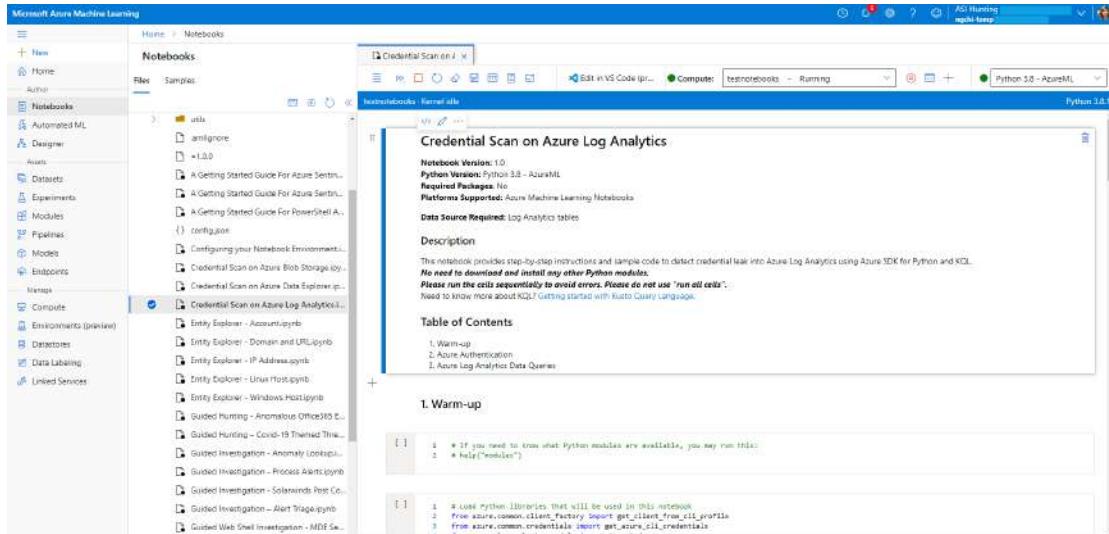
Use Microsoft Sentinel's powerful hunting search-and-query tools, based on the MITRE framework, which enable you to proactively hunt for security threats across your organization's data sources, before an alert is triggered. After you discover which hunting query provides high-value insights into possible attacks, you can also create custom detection rules based on your query, and surface those insights as alerts to your security incident responders. While hunting, you can create bookmarks for interesting events, enabling you to return to them later, share them with others, and group them with other correlating events to create a compelling incident for investigation.

The screenshot shows the Microsoft Sentinel Hunting interface. On the left, there's a navigation sidebar with options like Home, Cases, Dashboards, User profiles, and Hunting. The main area has tabs for General, Overview, Log, Threat management, Cases, Dashboards, User profiles, and Hunting. Under the Hunting tab, there's a search bar and a summary section showing "19 Total Queries" and "106 Total Results". Below this is a table of hunting queries, each with a star icon, a title, a description, provider (Microsoft), data source (SecurityEvent), and tactics. Some queries include OfficeActivity, SigninLogs, SecurityEvent, and PowerShell. To the right, there's a panel titled "New processes observed in last 24 hours" showing a table with Microsoft Provider, 103 Results, and SecurityEvent Data Source. It includes a description of the query results and a "View query results" link. At the bottom right is a "Run Query" button.

## Notebooks

Microsoft Sentinel supports Jupyter notebooks in Azure Machine Learning workspaces, including full libraries for machine learning, visualization, and data analysis.

Use notebooks in Microsoft Sentinel to extend the scope of what you can do with Microsoft Sentinel data. For example, perform analytics that aren't built in to Microsoft Sentinel, such as some Python machine learning features, create data visualizations that aren't built in to Microsoft Sentinel, such as custom timelines and process trees, or integrate data sources outside of Microsoft Sentinel, such as an on-premises data set.



- Microsoft Sentinel notebooks are intended for threat hunters or Tier 2-3 analysts, incident investigators, data scientists, and security researchers.
- Notebooks provide queries to both Microsoft Sentinel and external data, features for data enrichment, investigation, visualization, hunting, machine learning, and big data analytics.
- Notebooks are best for more complex chains of repeatable tasks, ad-hoc procedural controls, machine learning and custom analysis, support rich Python libraries for manipulating and visualizing data, and are useful in documenting and sharing analysis evidence.
- Notebooks require a higher learning curve and coding knowledge, and have limited automation support.

## Best practices for Microsoft Sentinel

### Regular SOC activities to perform

Schedule the following Microsoft Sentinel activities regularly to ensure continued security best practices:

#### Daily tasks

- Triage and investigate incidents. Review the Microsoft Sentinel Incidents page to check for new incidents generated by the currently configured analytics rules, and start investigating any new incidents.

- Explore hunting queries and bookmarks. Explore results for all built-in queries, and update existing hunting queries and bookmarks. Manually generate new incidents or update old incidents if applicable.
- Analytic rules. Review and enable new analytics rules as applicable, including both newly released or newly available rules from recently connected data connectors.
- Data connectors. Review the status, date, and time of the last log received from each data connector to ensure that data is flowing. Check for new connectors, and review ingestion to ensure set limits haven't been exceeded.
- Log Analytics Agent. Verify that servers and workstations are actively connected to the workspace, and troubleshoot and remediate any failed connections.
- Playbook failures. Verify playbook run statuses and troubleshoot any failures.

### **Weekly tasks**

- Workbook updates. Verify whether any workbooks have updates that need to be installed.
- Microsoft Sentinel GitHub repository review. Review the Microsoft Sentinel GitHub repository to explore whether there are any new or updated resources of value for your environment, such as analytics rules, workbooks, hunting queries, or playbooks.
- Microsoft Sentinel auditing. Review Microsoft Sentinel activity to see who has updated or deleted resources, such as analytics rules, bookmarks, and so on.

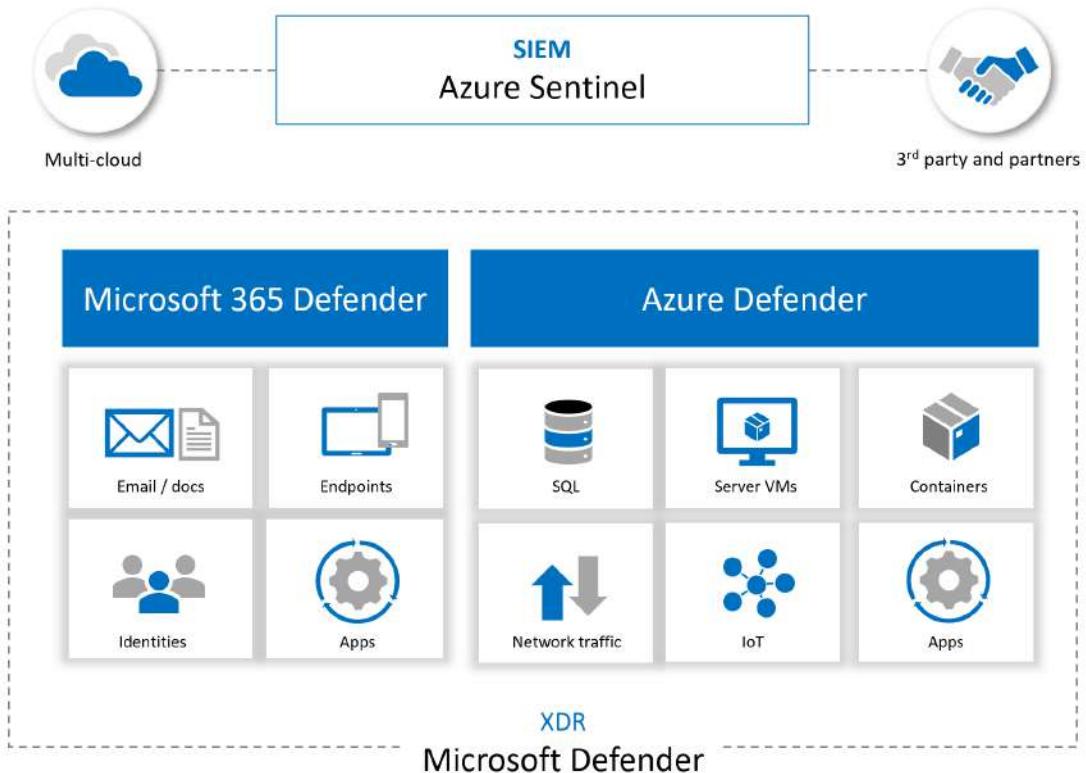
### **Monthly tasks**

- Review user access. Review permissions for your users and check for inactive users.
- Log Analytics workspace review. Review that the Log Analytics workspace data retention policy still aligns with your organization's policy.

### **Integrate with Microsoft security services**

Microsoft Sentinel is empowered by the components that send data to your workspace, and is made stronger through integrations with other Microsoft services. Any logs ingested into products such as Microsoft Defender for Cloud Apps, Microsoft Defender for Endpoint, and Microsoft Defender for Identity allow these services to create detections, and in turn provide those detections to Microsoft Sentinel. Logs can also be ingested directly into Microsoft Sentinel to provide a fuller picture for events and incidents.

For example, the following image shows how Microsoft Sentinel ingests data from other Microsoft services and multi-cloud and partner platforms to provide coverage for your environment:



More than ingesting alerts and logs from other sources, Microsoft Sentinel also:

- Uses the information it ingests with [machine learning](#) that allows for better event correlation, alert aggregation, anomaly detection, and more.
- Builds and presents interactive visuals via [workbooks](#), showing trends, related information, and key data used for both admin tasks and investigations.
- Runs [playbooks](#) to act on alerts, gathering information, performing actions on items, and sending notifications to various platforms.
- Integrates with partner platforms, such as ServiceNow and Jira, to provide essential services for SOC teams.
- Ingests and fetches enrichment feeds from [threat intelligence platforms](#) to bring valuable data for investigating.

## Manage and respond to incidents

The following image shows recommended steps in an incident management and response process.



The following sections provide high-level descriptions for how to use Microsoft Sentinel features for incident management and response throughout the process.

## Use the Incidents page and the Investigation graph

Start any triage process for new incidents on the Microsoft Sentinel Incidents page in Microsoft Sentinel and the Investigation graph.

Discover key entities, such as accounts, URLs, IP address, host names, activities, timeline, and more. Use this data to understand whether you have a false positive on hand, in which case you can close the incident directly.

Any generated incidents are displayed on the Incidents page, which serves as the central location for triage and early investigation. The Incidents page lists the title, severity, and related alerts, logs, and any entities of interest. Incidents also provide a quick jump into collected logs and any tools related to the incident.

The Incidents page works together with the Investigation graph, an interactive tool that allows users to explore and dive deep into an alert to show the full scope of an attack. Users can then construct a timeline of events and discover the extent of a threat chain.

If you discover that the incident is a true positive, take action directly from the Incidents page to investigate logs, entities, and explore the threat chain. After you've identified the threat and created a plan of action, use other tools in Microsoft Sentinel and other Microsoft security services to continue investigating.

## Handle incidents with workbooks

In addition to visualizing and displaying information and trends, Microsoft Sentinel workbooks are valuable investigative tools.

For example, use the Investigation Insights workbook to investigate specific incidents together with any associated entities and alerts. This workbook enables you to dive deeper into entities by showing related logs, actions, and alerts.

## Handle incidents with threat hunting

While investigating and searching for root causes, run built-in threat hunting queries and check results for any indicators of compromise.

During an investigation, or after having taken steps to remediate and eradicate the threat, use livestream to monitor, in real time, whether there are any lingering malicious events, or if malicious events are still continuing.

## Handle incidents with entity behavior

Entity behavior in Microsoft Sentinel allows users to review and investigate actions and alerts for specific entities, such as investigating into accounts and host names.

## Handle incidents with watchlists and threat intelligence

To maximize threat intelligence-based detections, make sure to use threat intelligence data connectors to ingest indicators of compromise:

- Connect data sources required by the Fusion and TI Map alerts
- Ingest indicators from TAXII and TIP platforms

Use indicators of compromise in analytics rules, when threat hunting, investigating logs, or generating more incidents.

Use a watchlist that combines data from ingested data and external sources, such as enrichment data. For example, create lists of IP address ranges used by your organization or recently terminated employees. Use watchlists with playbooks to gather enrichment data, such as adding malicious IP addresses to watchlists to use during detection, threat hunting, and investigations.

During an incident, use watchlists to contain investigation data, and then delete them when your investigation is done to ensure that sensitive data does not remain in view.

# DDoS Protection

## Azure DDoS Protection

Distributed denial of service (DDoS) attacks are some of the largest availability and security concerns facing customers that are moving their applications to the cloud. A DDoS attack attempts to exhaust an application's resources, making the application unavailable to legitimate users. DDoS attacks can be targeted at any endpoint that is publicly reachable through the internet.



Every property in Azure is protected by Azure's infrastructure DDoS (Basic) Protection at no additional cost. The scale and capacity of the globally deployed Azure network provides defense against common network-layer attacks through always-on traffic

monitoring and real-time mitigation. DDoS Protection Basic requires no user configuration or application changes. DDoS Protection Basic helps protect all Azure services, including PaaS services like Azure DNS.

Azure DDoS Protection Standard, combined with application design best practices, provides enhanced DDoS mitigation features to defend against DDoS attacks. It is automatically tuned to help protect your specific Azure resources in a virtual network. Protection is simple to enable on any new or existing virtual network, and it requires no application or resource changes. It has several advantages over the basic service, including logging, alerting, and telemetry.

Feature	DDoS Protection Basic	DDoS Protection Standard
Active traffic monitoring & always on detection	●	●
Automatic attack mitigations	●	●
Availability guarantee	●	●
Cost Protection	●	●
Mitigation policies tuned to customers application	●	●
Metrics & alerts	●	●
Mitigation reports	●	●
Mitigation flow logs	●	●
DDoS rapid response support		●

Azure DDoS protection does not store customer data.

## Features

- Native platform integration: Natively integrated into Azure. Includes configuration through the Azure portal. DDoS Protection Standard understands your resources and resource configuration.
- Turnkey protection: Simplified configuration immediately protects all resources on a virtual network as soon as DDoS Protection Standard is enabled. No intervention or user definition is required.
- Always-on traffic monitoring: Your application traffic patterns are monitored 24 hours a day, 7 days a week, looking for indicators of DDoS attacks. DDoS Protection Standard instantly and automatically mitigates the attack, once it is detected.
- Adaptive tuning: Intelligent traffic profiling learns your application's traffic over time, and selects and updates the profile that is the most suitable for your service. The profile adjusts as traffic changes over time.
- Multi-Layered protection: When deployed with a web application firewall (WAF), DDoS Protection Standard protects both at the network layer (Layer 3 and 4, offered by Azure DDoS Protection Standard) and at the application layer (Layer 7, offered by a WAF). WAF offerings include Azure Application Gateway WAF SKU as well as third-party web application firewall offerings available in the Azure Marketplace.
- Extensive mitigation scale: Over 60 different attack types can be mitigated, with global capacity, to protect against the largest known DDoS attacks.

- Attack analytics: Get detailed reports in five-minute increments during an attack, and a complete summary after the attack ends. Stream mitigation flow logs to Microsoft Sentinel or an offline security information and event management (SIEM) system for near real-time monitoring during an attack.
- Attack metrics: Summarized metrics from each attack are accessible through Azure Monitor.
- Attack alerting: Alerts can be configured at the start and stop of an attack, and over the attack's duration, using built-in attack metrics. Alerts integrate into your operational software like Microsoft Azure Monitor logs, Splunk, Azure Storage, Email, and the Azure portal.
- DDoS Rapid Response: Engage the DDoS Protection Rapid Response (DRR) team for help with attack investigation and analysis. To learn more, see [DDoS Rapid Response](#).
- Cost guarantee: Receive data-transfer and application scale-out service credit for resource costs incurred as a result of documented DDoS attacks.

## Pricing

DDoS protection plans have a fixed monthly charge of \$2,944 per month which covers up to 100 public IP addresses. Protection for additional resources will cost an additional \$30 per resource per month.

Under a tenant, a single DDoS protection plan can be used across multiple subscriptions, so there is no need to create more than one DDoS protection plan.

## Fundamental best practices

The following sections give prescriptive guidance to build DDoS-resilient services on Azure.

### Design for security

Ensure that security is a priority throughout the entire lifecycle of an application, from design and implementation to deployment and operations. Applications can have bugs that allow a relatively low volume of requests to use an inordinate amount of resources, resulting in a service outage.

To help protect a service running on Microsoft Azure, you should have a good understanding of your application architecture and focus on the five pillars of software quality. You should know typical traffic volumes, the connectivity model between the application and other applications, and the service endpoints that are exposed to the public internet.

Ensuring that an application is resilient enough to handle a denial of service that's targeted at the application itself is most important. Security and privacy are built into the Azure platform, beginning with the Security Development Lifecycle (SDL). The SDL addresses security at every development phase and ensures that Azure is continually updated to make it even more secure.

## Design for scalability

Scalability is how well a system can handle increased load. Design your applications to scale horizontally to meet the demand of an amplified load, specifically in the event of a DDoS attack. If your application depends on a single instance of a service, it creates a single point of failure. Provisioning multiple instances makes your system more resilient and more scalable.

For Azure App Service, select an App Service plan that offers multiple instances. For Azure Cloud Services, configure each of your roles to use multiple instances. For Azure Virtual Machines, ensure that your virtual machine (VM) architecture includes more than one VM and that each VM is included in an availability set. We recommend using virtual machine scale sets for auto scaling capabilities.

## Defense in depth

The idea behind defense in depth is to manage risk by using diverse defensive strategies. Layering security defenses in an application reduces the chance of a successful attack. We recommend that you implement secure designs for your applications by using the built-in capabilities of the Azure platform.

For example, the risk of attack increases with the size (surface area) of the application. You can reduce the surface area by using an approval list to close down the exposed IP address space and listening ports that are not needed on the load balancers (Azure Load Balancer and Azure Application Gateway). Network security groups (NSGs) are another way to reduce the attack surface. You can use service tags and application security groups to minimize complexity for creating security rules and configuring network security, as a natural extension of an application's structure.

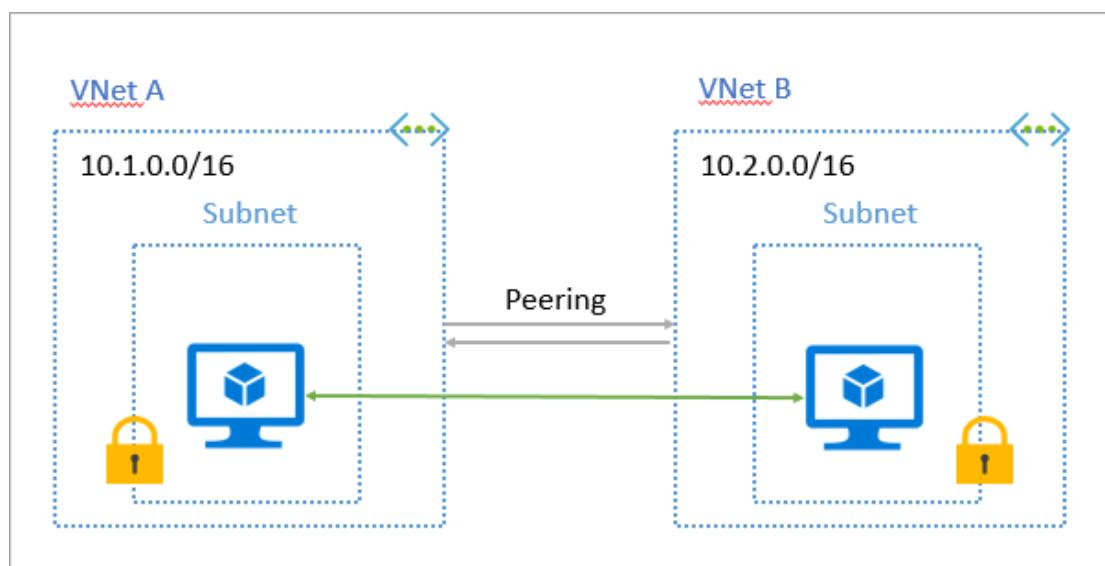
You should deploy Azure services in a virtual network whenever possible. This practice allows service resources to communicate through private IP addresses. Azure service traffic from a virtual network uses public IP addresses as source IP addresses by default. Using service endpoints will switch service traffic to use virtual network private addresses as the source IP addresses when they're accessing the Azure service from a virtual network.

We often see customers' on-premises resources getting attacked along with their resources in Azure. If you're connecting an on-premises environment to Azure, we recommend that you minimize exposure of on-premises resources to the public internet. You

can use the scale and advanced DDoS protection capabilities of Azure by deploying your well-known public entities in Azure. Because these publicly accessible entities are often a target for DDoS attacks, putting them in Azure reduces the impact on your on-premises resources.

## Dedicated HSM

### What is Azure Dedicated HSM?



Azure Dedicated HSM is an Azure service that provides cryptographic key storage in Azure. Dedicated HSM meets the most stringent security requirements. It's the ideal solution for customers who require FIPS 140-2 Level 3-validated devices and complete and exclusive control of the HSM appliance.

HSM devices are deployed globally across several Azure regions. They can be easily provisioned as a pair of devices and configured for high availability. HSM devices can also be provisioned across regions to assure against regional-level failover. Microsoft delivers the Dedicated HSM service by using the Thales Luna 7 HSM model A790 appliances. This device offers the highest levels of performance and cryptographic integration options.

After they're provisioned, HSM devices are connected directly to a customer's virtual network. They can also be accessed by on-premises application and management tools when you configure point-to-site or site-to-site VPN connectivity. Customers get the software and documentation to configure and manage HSM devices from Thales customer support portal.

## Why use Azure Dedicated HSM?

### FIPS 140-2 Level-3 compliance

Many organizations have stringent industry regulations that dictate that cryptographic keys must be stored in FIPS 140-2 Level-3 validated HSMs. Azure Dedicated HSM and a new single-tenant offering, Azure Key Vault Managed HSM, help customers from various industry segments, such as financial services industry, government agencies, and others meet FIPS 140-2 Level-3 requirements. While Microsoft's multi-tenant Azure Key Vault service currently uses FIPS 140-2 Level-2 validated HSMs.

### Single-tenant devices

Many customers have a requirement for single tenancy of the cryptographic storage device. The Azure Dedicated HSM service enables them to provision a physical device from one of Microsoft's globally distributed datacenters. After it's provisioned to a customer, only that customer can access the device.

### Full administrative control

Many customers require full administrative control and sole access to their device for administrative purposes. After a device is provisioned, only the customer has administrative or application-level access to the device.

Microsoft has no administrative control after the customer accesses the device for the first time, at which point the customer changes the password. From that point, the customer is a true single-tenant with full administrative control and application-management capability. Microsoft does maintain monitor-level access (not an admin role) for telemetry via serial port connection. This access covers hardware monitors such as temperature, power supply health, and fan health.

The customer is free to disable this monitoring needed. However, if they disable it, they won't receive proactive health alerts from Microsoft.

### High performance

The Thales device was selected for this service for a variety of reasons. It offers a broad range of cryptographic algorithm support, a variety of supported operating systems, and broad API support. The specific model that's deployed offers excellent performance with 10,000 operations per second for RSA-2048. It supports 10 partitions that can be used for unique application instances. This device is a low latency, high capacity, and high throughput device.

## **Unique cloud-based offering**

Microsoft recognized a specific need for a unique set of customers. It is the only cloud provider that offers new customers a dedicated HSM service that is FIPS 140-2 Level 3-validated and offers such an extent of cloud-based and on-premises application integration.

## **Is Azure Dedicated HSM right for you?**

Azure Dedicated HSM is a specialized service that addresses unique requirements for a specific type of large-scale organization. As a result, it's expected that the bulk of Azure customers will not fit the profile of use for this service. Many will find the Azure Key Vault service to be more appropriate and cost effective. To help you decide if it's a fit for your requirements, we've identified the following criteria.

### **Best fit**

Azure Dedicated HSM is most suitable for “lift-and-shift” scenarios that require direct and sole access to HSM devices. Examples include:

- Migrating applications from on-premises to Azure Virtual Machines
- Migrating applications from Amazon AWS EC2 to virtual machines that use the AWS Cloud HSM Classic service (Amazon is not offering this service to new customers)
- Running shrink-wrapped software such as Apache/Nginx SSL Offload, Oracle TDE, and ADCS in Azure Virtual Machines

### **Not a fit**

Azure Dedicated HSM is not a good fit for the following type of scenario: Microsoft cloud services that support encryption with customer-managed keys (such as Azure Information Protection, Azure Disk Encryption, Azure Data Lake Store, Azure Storage, Azure SQL Database, and Customer Key for Office 365) that are not integrated with Azure Dedicated HSM.

### **It depends**

Whether Azure Dedicated HSM will work for you depends on a potentially complex mix of requirements and compromises that you can or cannot make. An example is the FIPS 140-2 Level 3 requirement. This requirement is common, and Azure Dedicated HSM and a new single-tenant offering, Azure Key Vault Managed HSM are currently the only options for meeting it. If these mandated requirements aren't relevant, then often it's a choice between Azure Key Vault and Azure Dedicated HSM. Assess your requirements before making a decision.

Situations in which you will have to weigh your options include:

- New code running in a customer's Azure virtual machine
- SQL Server TDE in an Azure virtual machine
- Azure Storage client-side encryption
- SQL Server and Azure SQL DB Always Encrypted

## Azure Dedicated HSM high availability

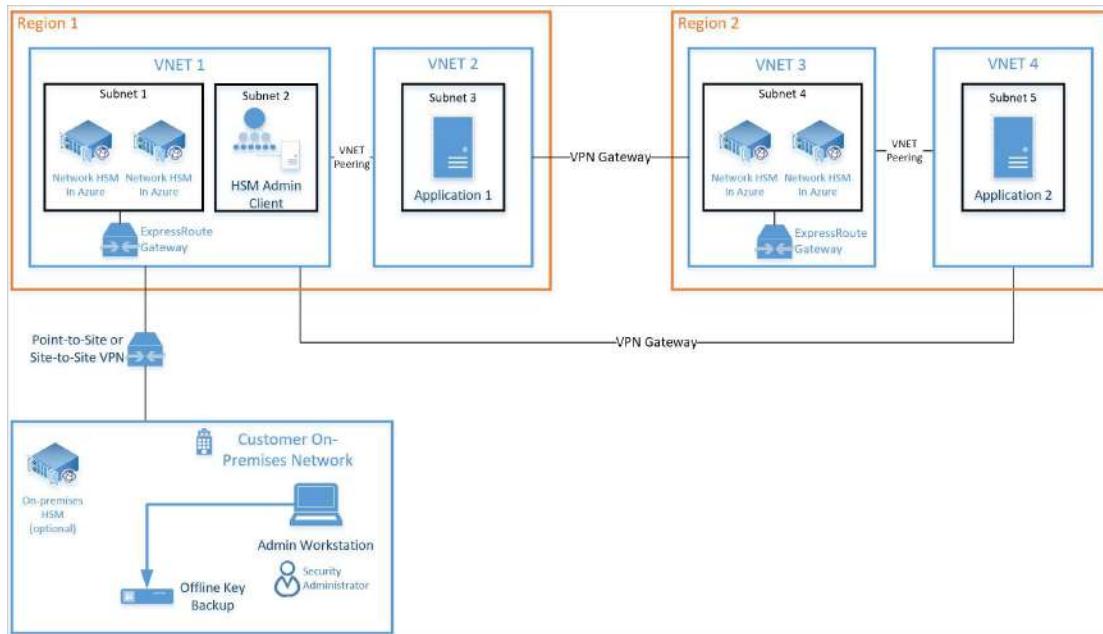
Azure Dedicated HSM is underpinned by Microsoft's highly available datacenters. However, any highly available datacenter is vulnerable to localized failures and in extreme circumstances, regional level failures. Microsoft deploys HSM devices in different datacenters within a region to ensure provisioning multiple devices does not lead to those devices sharing a single rack.

A further level of high availability can be achieved by pairing these HSMs across the datacenters in a region using the Thales HA Group feature. It is also possible to pair devices across regions to address regional failover in a disaster recovery situation. With this multi-layered high availability configuration, any device failure will be automatically addressed to keep applications working. All data centers also have spare devices and components on-site so any failed device can be replaced in a timely fashion.

### High availability example

Information on how to configure HSM devices for high availability at the software level is in the 'Thales Luna 7 HSM Administration Guide'. This document is available at the Thales HSM Page.

The following diagram shows a highly available architecture. It uses multiple devices in a region and multiple devices paired in a separate region. This architecture uses a minimum of four HSM devices and virtual networking components.



## Azure Dedicated HSM monitoring

The Azure Dedicated HSM Service provides a physical device for sole customer use with complete administrative control and management responsibility. The device made available is a [Thales Luna 7 HSM model A790](#). Microsoft will have no administrative access once provisioned by a customer, beyond physical serial port attachment as a monitoring role. As a result, customers are responsible for typical operational activities including comprehensive monitoring and log analysis. Customers are fully responsible for applications that use the HSMs and should work with Thales for support or consulting assistance.

## Microsoft monitoring

The Thales Luna 7 HSM device in use has by default SNMP and serial port as options for monitoring the device. Microsoft has used the serial port connection as a physical means to connect to the device to retrieve basic telemetry on device health. This includes items such as temperature and component status such as power supplies and fans. To achieve this, Microsoft uses a non-administrative “monitor” role set up on the Thales device. This role gives the ability to retrieve the telemetry but does not give any access to the device in terms of administrative task or in any way viewing cryptographic information.

Our customers can be assured their device is truly their own to manage, administer, and use for sensitive cryptographic key storage. In case any customer is not satisfied with this minimal access for basic health monitoring, they do have the option to disable the monitoring account. The obvious consequence of this is that Microsoft will have no information and hence no ability to provide any proactive notification of device health issues. In this situation, the customer is responsible for the health of the device. The monitor function itself is set up to poll the device every 10 minutes to get health data. Due to the error prone nature of serial communications, only after multiple negative health indicators over a one hour period would an alert be raised. This alert would ultimately lead to proactive customer communication notifying the issue. Depending on the nature of the issue, the appropriate course of action would be taken to reduce impact and ensure low risk remediation.

For example, a power supply failure is a hot-swap procedure with no resultant tamper event so can be performed with low impact and minimal risk to operation. Other procedures may require a device to be zeroized and deprovisioned to minimize any security risk to the customer. In this situation a customer would provision an alternate device, rejoin a high availability pairing thus triggering device synchronization. Normal operation would resume in minimal time, with minimal disruption and lowest security risk.

## **Customer monitoring**

A value proposition of the Dedicated HSM service is the control the customer gets of the device, especially considering it is a cloud delivered device. A consequence of this control is the responsibility to monitor and manage the health of the device. The Thales Luna 7 HSM device comes with guidance for SNMP and Syslog implementation. Customers of the Dedicated HSM service are recommended to use this even when the Microsoft monitor account remains active and should consider it mandatory if they disable the Microsoft monitor account. Either technique available would allow a customer to identify issues and call Microsoft support to initiate appropriate remediation work.

## **Azure Dedicated HSM networking**

Azure Dedicated HSM requires a highly secure networking environment. This is true whether it is from the Azure cloud back to the customer's IT environment (on-premises), using distributed applications or for high availability scenarios. Azure Networking provides this and there are four distinct areas that must be addressed.

- Creating HSM devices inside your Virtual Network (VNet) in Azure
- Connecting on-premises to cloud-based resources for the configuration and management of HSM devices
- Creating and connecting virtual networks for inter-connecting application resources and HSM devices
- Connecting virtual networks across regions for inter-communication and also to enable high availability scenarios

## **Virtual network for your Dedicated HSMs**

Dedicated HSMs are integrated into a Virtual Network and placed in the customers own private network in Azure. This enables access to the devices from virtual machines or compute resources in the virtual network.

## **Virtual networks**

Before provisioning a Dedicated HSM device, customers will first need to create a Virtual Network in Azure or use one that already exists in the customers subscription. The virtual network defines the security perimeter for the Dedicated HSM device.

## **Subnets**

Subnets segment the virtual network into separate address spaces usable by the Azure resources you place in them. Dedicated HSMs are deployed into a subnet in the virtual network. Each Dedicated HSM device that is deployed in the customer's subnet will receive a private IP address from this subnet. The subnet in which the HSM device is deployed needs to be explicitly delegated to the service: Microsoft.HardwareSecurityModules/dedicatedHSMs. This grants certain permissions to the HSM service for deployment into the subnet. Delegation to Dedicated HSMs imposes certain policy restrictions on the subnet.

Network Security Groups (NSGs) and User-Defined Routes (UDRs) are currently not supported on delegated subnets. As a result, once a subnet is delegated to dedicated HSMs, it can only be used to deploy HSM resources. Deployment of any other customer resources into the subnet will fail. There is no requirement on how large or small the subnet for Dedicated HSM should be, however each HSM device will consume one private IP, so it should be ensured the subnet is large enough to accommodate as many HSM devices as required for deployment.

### **ExpressRoute gateway**

A requirement of the current architecture is configuration of an ExpressRoute gateway in the customers subnet where an HSM device needs to be placed to enable integration of the HSM device into Azure. This ExpressRoute gateway cannot be utilized for connecting on-premises locations to the customers HSM devices in Azure.

### **Connecting your on-premises IT to Azure**

When creating cloud-based resources, it is a typical requirement for a private connection back to on-premises IT resources. In the case of Dedicated HSM, this will predominantly be for the HSM client software to configure the HSM devices and also for activities such as backups and pulling logs from HSMs for analysis. A key decision point here is the nature of the connection as there are options. The most flexible option is Site-to-Site VPN as there will likely be multiple on-premises resources that require secure communication with resources (including HSMs) in the Azure cloud. This will require a customer organization to have a VPN device to facilitate the connection. A Point-to-Site VPN connection can be used if there is only a single end-point on-premises such as a single administration workstation.

### **Point-to-Site VPN**

A point-to-site Virtual Private Network is the simplest form of secure connection to a single endpoint on-premises. This may be relevant if you intend to only have a single administration workstation for Azure-based dedicated HSMs.

### **Site-to-Site VPN**

A site-to-site Virtual Private Network allows for secure communication between Azure-based Dedicated HSMs and your on-premises IT. A reason to do this is having a backup facility for the HSM's on-premises and needing a connection between the two for running the backup.

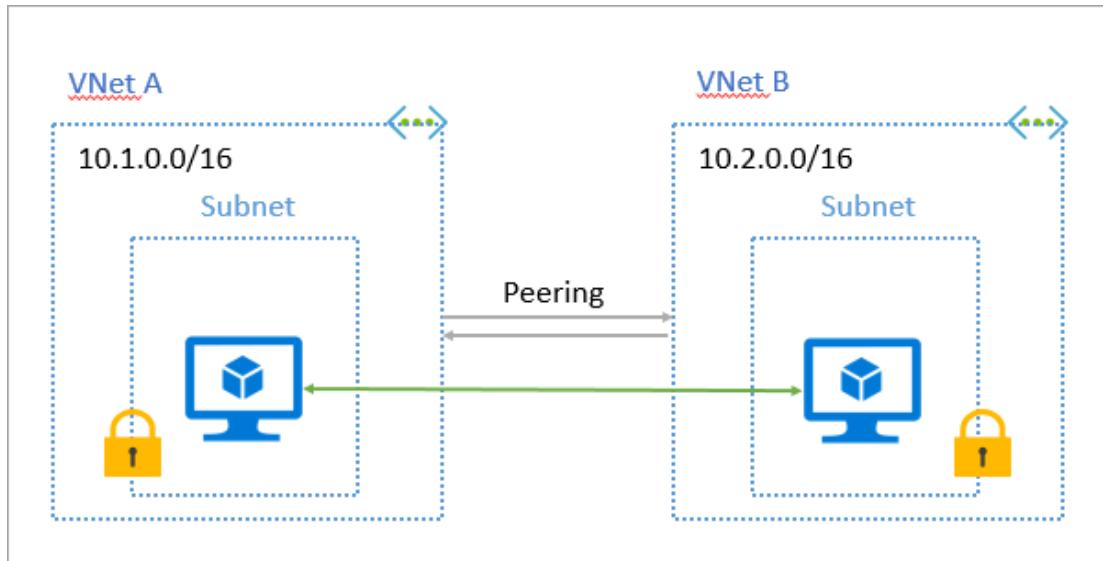
### **Connecting virtual networks**

A typical deployment architecture for Dedicated HSM will start with a single virtual network and corresponding subnet in which the HSM devices are created and

provisioned. Within that same region, there could well be additional virtual networks and subnets for application components that would make use of the Dedicated HSM. To enable communication across these networks, we use Virtual Network Peering.

## Virtual network peering

When there are multiple virtual networks within a region that need to access each other's resources, Virtual Network Peering can be used to create secure communication channels between them. Virtual network peering provides not only secure communications but also ensures low-latency and high-bandwidth connections between the resources in Azure.

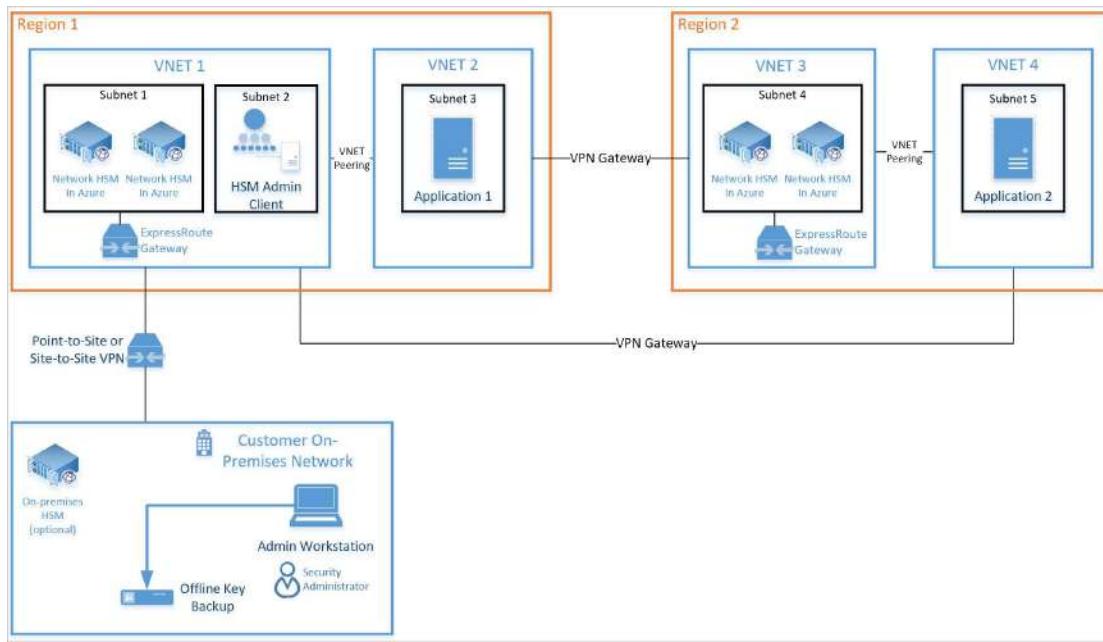


## Connecting across Azure Regions

The HSM devices have the ability, via software libraries, to redirect traffic to an alternate HSM. Traffic redirection is helpful if devices fail or access to a device is lost. Regional level failure scenarios can be mitigated by deploying HSMs in other regions and enabling communication between virtual networks across regions.

## Cross region HA using VPN gateway

For globally distributed applications or for high availability regional failover scenarios, it is required to connect virtual networks across regions. With Azure Dedicated HSM, high-availability can be achieved by using a VPN Gateway that provides a secure tunnel between the two virtual networks.



## Networking Restrictions

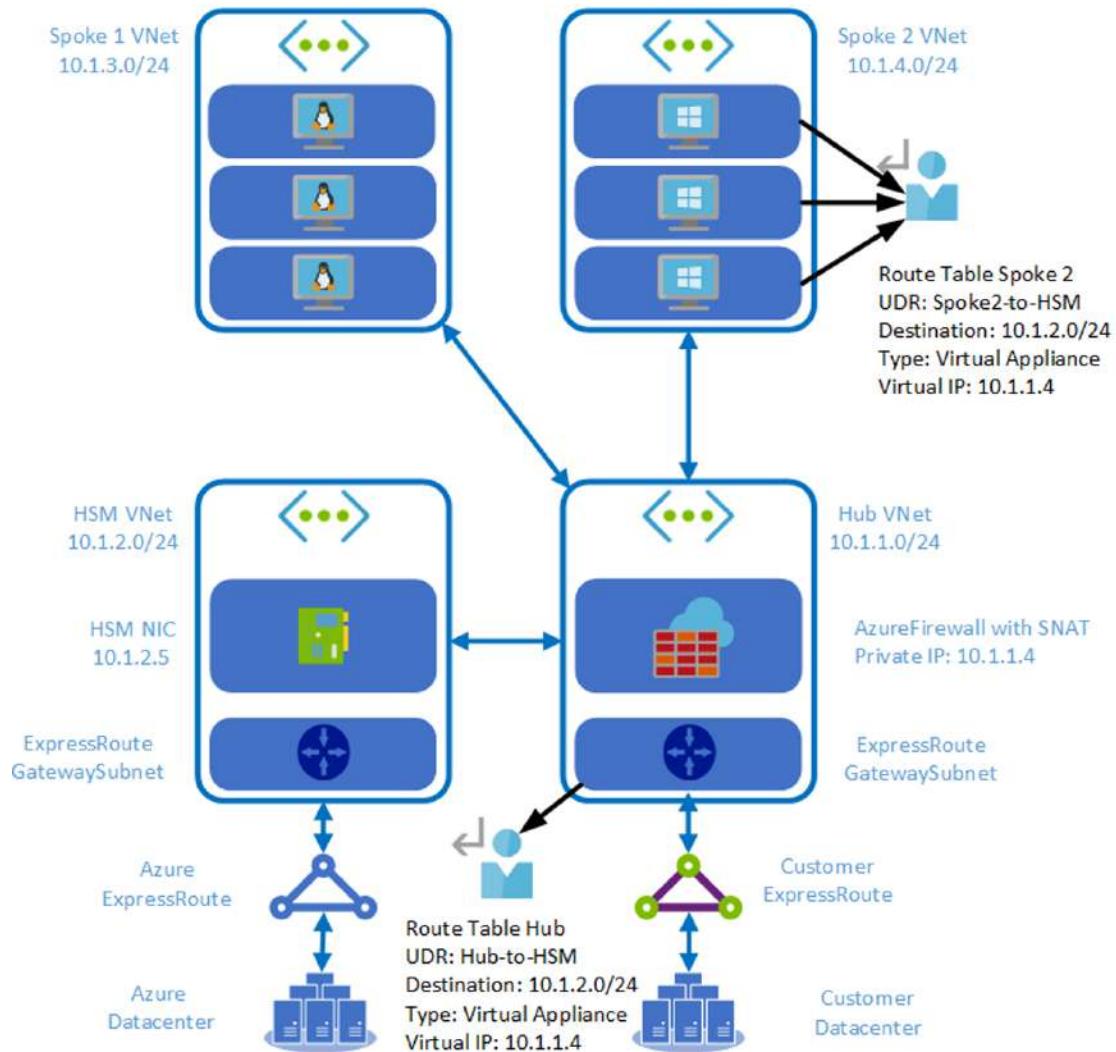
The HSM NIC which resides in the Dedicated HSM VNet cannot use Network Security Groups, or User Defined Routes. This means that it is not possible to set default-deny policies from the standpoint of the Dedicated HSM VNet, and that other network segments must be allowlisted to gain access to the Dedicated HSM service.

Adding the Network Virtual Appliances (NVA) Proxy solution also allows for an NVA firewall in the transit/DMZ hub to be logically placed in front of the HSM NIC, thus providing the needed alternative to NSGs and UDRs.

## Solution Architecture

This networking design requires the following elements:

1. A transit or DMZ hub VNet with an NVA proxy tier. Ideally two or more NVAs are present.
2. An ExpressRoute circuit with a private peering enabled and a connection to the transit hub VNet.
3. A VNet peering between the transit hub VNet and the Dedicated HSM VNet.
4. An NVA firewall or Azure Firewall can be deployed offer DMZ services in the hub as an option.
5. Additional workload spoke VNets can be peered to the hub VNet. The Gemalto client can access the dedicated HSM service through the hub VNet.



Since adding the NVA proxy solution also allows for an NVA firewall in the transit/DMZ hub to be logically placed in front of the HSM NIC, thus providing the needed default-deny policies. In our example, we will use the Azure Firewall for this purpose and will need the following elements in place:

1. An Azure Firewall deployed into subnet “AzureFirewallSubnet” in the DMZ hub VNet
2. A Routing Table with a UDR that directs traffic headed to the Azure ILB private endpoint into the Azure Firewall. This Routing Table will be applied to the GatewaySubnet where the customer [ExpressRoute Virtual Gateway](#) resides
3. Network security rules within the AzureFirewall to allow forwarding between a trusted source range and the Azure IBL private endpoint listening on TCP port 1792. This security logic will add the necessary “default deny” policy against the Dedicated HSM service. Meaning, only trusted source IP ranges will be allowed into the Dedicated HSM service. All other ranges will be dropped.
4. A Routing Table with a UDR that directs traffic headed to on-prem into the Azure Firewall. This Routing Table will be applied to the NVA proxy subnet.

5. An NSG applied to the Proxy NVA subnet to trust only the subnet range of the Azure Firewall as a source, and to only allow forwarding to the HSM NIC IP address over TCP port 1792.

## Alternative to UDRs

The NVA tier solution mentioned above works as an alternative to UDRs. There are some important points to note.

1. Network Address Translation should be configured on NVA to allow for return traffic to be routed correctly.
2. Customers should disable the client ip-check in Luna HSM configuration to use VNA for NAT. The following commands serve as an example.

Copy

Disable:

```
[hsm01] lunash:>ntls ipcheck disable  
NTLS client source IP validation disabled  
Command Result : 0 (Success)
```

Show:

```
[hsm01] lunash:>ntls ipcheck show  
NTLS client source IP validation : Disable  
Command Result : 0 (Success)
```

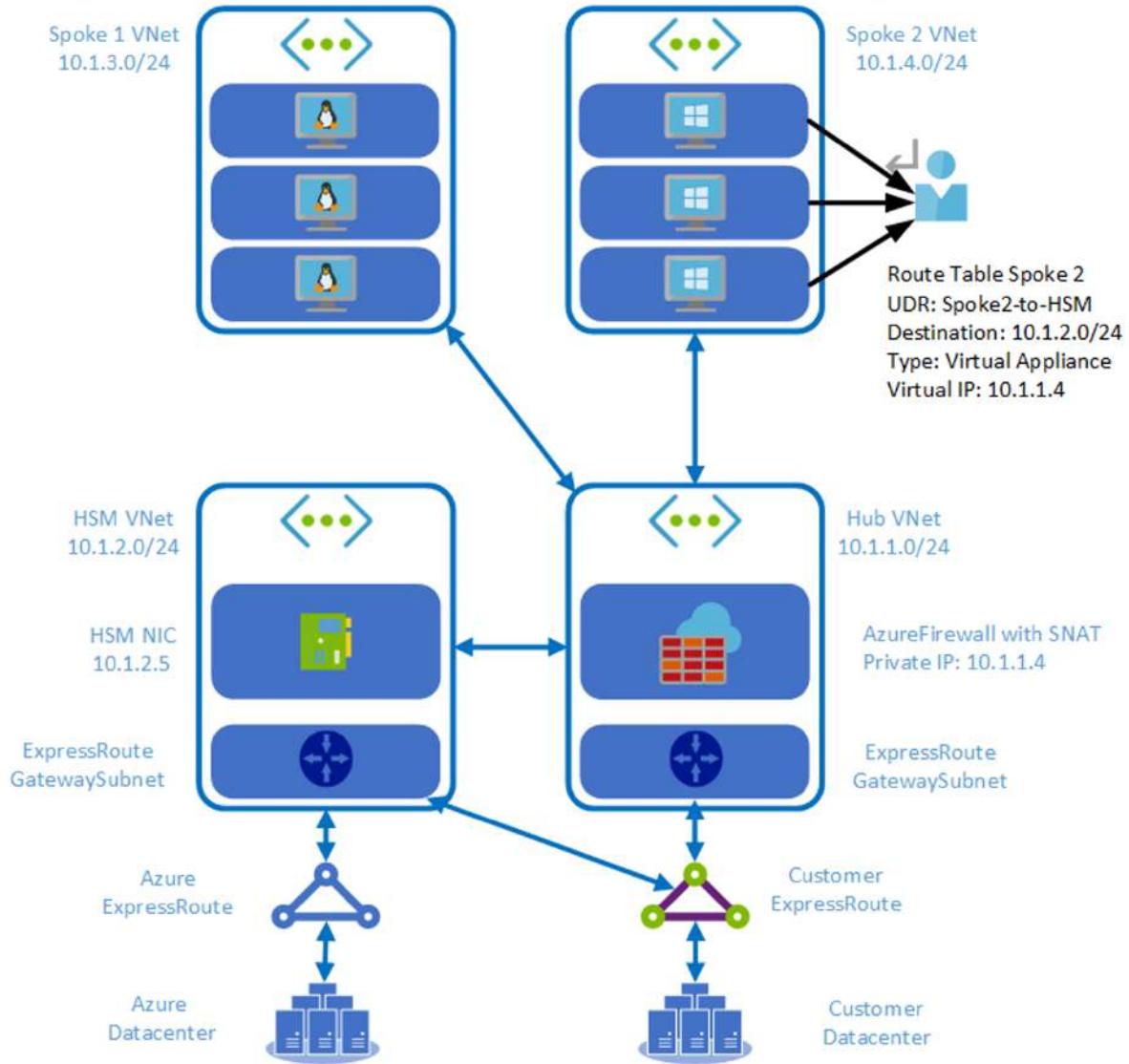
3. Deploy UDRs for ingress traffic into the NVA tier.
4. As per design, HSM subnets will not initiate an outbound connection request to the platform tier.

## Alternative to using Global VNET Peering

There are a couple of architectures you can use as an alternative to Global VNet peering.

1. Use Vnet-to-Vnet VPN Gateway Connection
2. Connect HSM VNET with another VNET with an ER circuit. This works best when a direct on-premises path is required or VPN VNET.

## HSM with direct Express Route connectivity



## Azure Dedicated HSM physical security

Azure Dedicated HSM helps you meet advanced security requirements for key storage. It is managed following stringent security practices throughout its full lifecycle to meet customers needs.

## Security through procurement

Microsoft follows a secure procurement process. We manage the chain of custody and ensure that the specific device ordered and shipped is the device arriving at our data centers. The devices are in serialized tamper-evident plastic bags and containers. They are stored in a secure storage area until commissioned in the data gallery of the data center. The racks containing the HSM devices are considered high business

impact(HBI). The devices are locked and under video surveillance at all times front and back.

## Security through deployment

HSMs are installed in racks together with associated networking components. Once installed, they must be configured before they are made available as part of the Azure Dedicated HSM Service. This configuration activity is performed by Microsoft employees that have undergone a background check. “Just In Time” (JIT) administration is used to limit access to only the right employees and for only the time that access is needed. The procedures and systems used also ensure that all activity related to the HSM devices is logged.

## Security in operations

HSMs are hardware appliances (the actual HSM being a PCI card within the appliance) so it is possible that component level issues may arise. Potential issues include but are not limited to fan and power supply failures. This type of event will require maintenance or break/fix activities to replace any swappable components.

## Component replacement

After a device is provisioned and under customer management, the hot-swappable power supply is the only component that would be replaced. This component is outside of the security boundary and does not cause a tamper event. A ticketing system is used to authorize a Microsoft engineer to access the rear of the HBI rack. When the ticket is processed a temporary physical key is issued. This key gives the engineer access to the device and allows them to swap the affected component. Any other access (that is, tamper event causing) would be done when a device is not allocated to a customer thus minimizing security and availability risk.

## Device replacement

In the event of total device failure, a process similar to the one used during component failure is followed. If a customer is not able to zeroize the device, or the device is in an unknown state, the data bearing devices will be removed and placed in an in-rack destruction bin. Devices placed in the bin will be destroyed in a controlled and secure manner. No data bearing devices from an HBI rack will leave a Microsoft datacenter.

## Other Rack Access Activities

If a Microsoft engineer must access the rack used by HSM devices (for example, networking device maintenance), standard security procedures will be used to gain access to the HBI secure rack. All access will be under video surveillance. The HSM

devices are validated to FIPS 140-2 Level 3 so any unauthorized access to the HSM Devices will be signaled to the customer and data will be zeroized.

## Logical level security considerations

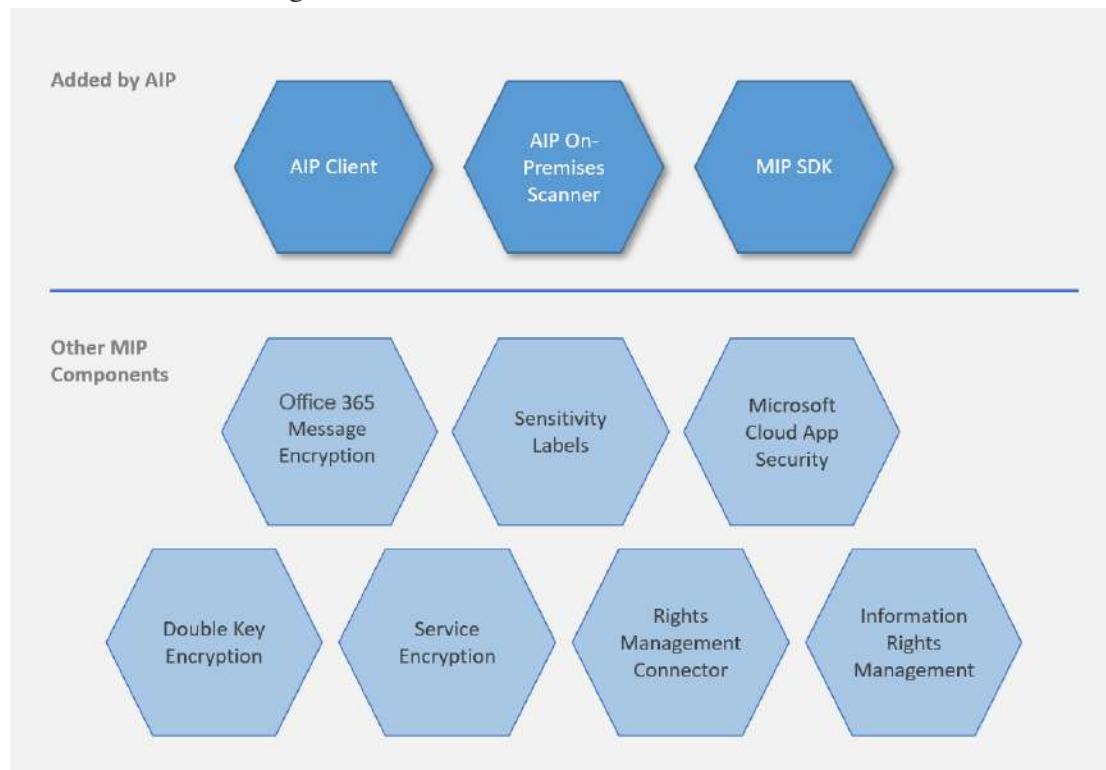
HSMs are provisioned to a virtual network created by the customer within the customer's private IP Address space. This configuration provides a valuable logical network level isolation and ensures access only by the customer. This implies that all logical level security controls are the responsibility of the customer.

# Information Protection

## What is Azure Information Protection?

Azure Information Protection (AIP) is a cloud-based solution that enables organizations to discover, classify, and protect documents and emails by applying labels to content. AIP is part of the Microsoft Information Protection (MIP) solution, and extends the labeling and classification functionality provided by Microsoft 365.

The following image shows the Azure Information Protection additions to MIP, including the unified labeling client, scanner, and SDK.

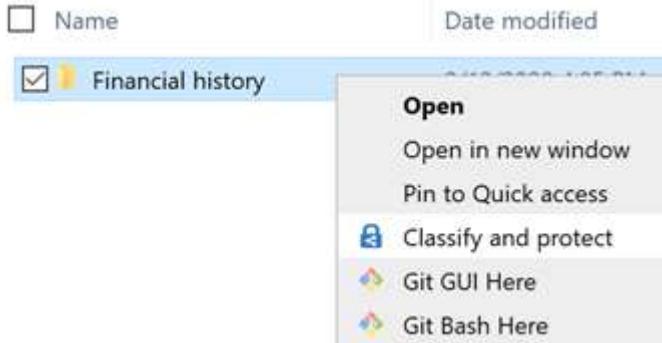


Microsoft Information Protection is the common information protection stack that's leveraged by AIP's unified labeling client.

## AIP unified labeling client

The Azure Information Protection unified labeling client extends labeling, classification, and protection capabilities to additional file types, as well as to the File Explorer and PowerShell.

For example, in the File Explorer, right-click one or more files and select Classify and protect to manage the AIP functionality on the selected files.



Download the client from the Microsoft Azure Information Protection download page.

## AIP on-premises scanner

The Azure Information Protection on-premises scanner enables administrators to scan their on-premises file repositories for sensitive content that must be labeled, classified, and/or protected.

The on-premises scanner is installed using PowerShell cmdlets provided as part of the unified labeling client, and can be managed using PowerShell and the Azure Information Protection area in the Azure portal.

For example, use the scanner data shown on the Azure portal to find repositories on your network that might have sensitive content at risk:

The screenshot shows the Azure Information Protection portal's 'Repositories (Preview)' section. On the left, there's a navigation sidebar with various links like General, Analytics, Classifications, Policies, Scanner, and Manage. The 'Repositories (Preview)' link is highlighted with a red box. The main area has a pie chart titled 'Repositories by status' showing 14 managed and 0 not managed. Below it is a table titled 'Top 10 unmanaged repositories by access' with columns for Path, Content Scan Job, Last Scan End Time, Discovered By, Effective Public Access, Repository Permissions, and Share Permissions. Each row in the table also has a red box highlighting the 'Content Scan Job' column, which shows values like 'Demo' or 'Unknown'.

Download the scanner installation together with the client from the Microsoft Azure Information Protection download page.

## **Microsoft Information Protection SDK**

The Microsoft Information Protection SDK extends sensitivity labels to third-party apps and services. Developers can use the SDK to build built-in support for applying labels and protection to files.

For example, you might use the MIP SDK for:

- A line-of-business application that applies classification labels to files on export.
- A CAD/CAM design application provides built-in support for Microsoft Information Protection labeling.
- A cloud access security broker or data loss prevention solution reasons over data encrypted with Azure Information Protection.

## **Azure security baseline for Azure Information Protection**

### **Network Security**

#### **NS-1: Implement security for internal traffic**

Guidance: Not applicable; there are no specific actions, which need to be taken by the customer to secure the internal traffic for Microsoft Azure Information Protection. The underlying infrastructure for secure internal traffic is fully managed by Microsoft.

To enable backwards compatibility, Azure Information Protection supports communicating on older versions of TSL, such as 1.0, for inbound traffic. However, all outbound requests use TLS 1.2 or newer. We recommend that customers use TLS 1.2.

Responsibility: Microsoft

Microsoft Defender for Cloud monitoring: None

#### **NS-6: Simplify network security rules**

Guidance: Use Virtual Network service tags to define network access controls on network security groups or Azure Firewall, which is configured for your Azure Information Protection resources.

When creating security rules, use service tags in place of specific IP addresses. Specify the service tag name, such as {AzureInformationProtection}, in the appropriate source or destination field of a rule, to allow or deny the traffic for the corresponding service.

Microsoft manages the address prefixes encompassed by the service tag, and automatically updates the service tag as addresses change.

Responsibility: Customer  
Microsoft Defender for Cloud monitoring: None

## Identity Management

### IM-1: Standardize Azure Active Directory as the central identity and authentication system

Guidance: Azure Information Protection is integrated with Azure Active Directory (Azure AD), which is Azure's default identity and access management service. Make it a high priority to secure Azure AD in your organization's cloud security practice.

Review the Azure AD identity secure score to help you assess your identity security posture relative to Microsoft's best practice recommendations. Use the score to gauge how closely your configuration matches best practice recommendations, and to make improvements in your security posture.

Standardize Azure AD to govern your organization's identity and access management in:

- Microsoft Cloud resources, such as the Azure portal, Azure Storage, Azure Virtual Machines (Linux and Windows), Azure Key Vault, Platform as a Service (PaaS), and Software as a Service (SaaS) applications
- Your organization's resources, such as applications on Azure or your corporate network resources

Azure AD supports external identities to allow users without a Microsoft account to sign in to their applications and resources with their non-Microsoft accounts.

Responsibility: Customer

Microsoft Defender for Cloud monitoring: None

### IM-2: Manage application identities securely and automatically

Guidance: Azure Information Protection is integrated with Azure Active Directory (Azure AD), which is Azure's identity and access management service. Azure Rights Management service uses an Azure AD application identity while accessing customers' keys stored with Azure Key Vault for Bring Your Own Key (BYOK) scenarios. Authorizing Azure Rights Management service to access your keys is achieved through configuring Azure Key Vault access policies, which can be done either using the Azure portal or using PowerShell.

Responsibility: Customer

Microsoft Defender for Cloud monitoring: None

### IM-3: Use Azure AD single sign-on (SSO) for application access

Guidance: Azure Information Protection is integrated with Azure Active Directory (Azure AD), which is Azure's default identity and access management service.

Azure Information Protection uses Azure AD to provide identity and access management to Azure resources, cloud applications, and on-premises applications. This includes enterprise identities such as employees, as well as external identities such as partners, vendors, and suppliers. This enables single sign-on to manage and secure access to your organization's data and resources on-premises and in the cloud. Connect all your users, applications, and devices to the Azure AD for seamless, secure access and greater visibility and control.

Responsibility: Customer

Microsoft Defender for Cloud monitoring: None

## Privileged Access

### PA-1: Protect and limit highly privileged users

Guidance: Azure Information Protection is integrated with Azure Active Directory (Azure AD), which is Azure's default identity and access management service.

Azure Information Protection includes an administrator-level role in Azure AD. Users assigned to the Administrator role have full permissions in the Azure Information Protection service. Administrator role can be used to configure labels for the Azure Information Protection policy, managing protection templates, and activating protection. However, the Administrator role does not grant any permissions in Identity Protection Center, Privileged Identity Management, Monitor Microsoft 365 Service Health, or Office 365 Security & Compliance Center.

Limit the number of highly privileged accounts or roles and protect these accounts at an elevated level, as users with this privilege can directly or indirectly read and modify every resource in your Azure environment. Enable just-in-time (JIT) privileged access to Azure resources and Azure AD using Privileged Identity Management (PIM). Just-in-time access grants temporary permissions to perform privileged tasks only when users need it. PIM can also generate security alerts when there is suspicious or unsafe activity in your Azure AD organization.

Responsibility: Customer

Microsoft Defender for Cloud monitoring: None

### PA-3: Review and reconcile user access regularly

Guidance: Azure Information Protection is integrated with Azure Active Directory (Azure AD), which is Azure's default identity and access management service.

Use Azure AD to manage resources, review user accounts, and access assignments regularly to ensure that the accounts and their access are valid. Conduct Azure AD access reviews to review group memberships, access to enterprise applications, and role assignments. Discover stale accounts with Azure AD reporting. Azure AD's

Privileged Identity Management features can be used to create access review report workflow to facilitate the review process.

In addition, Azure Privileged Identity Management can also be configured to alert when an excessive number of administrator accounts are created, and to identify administrator accounts that are stale or improperly configured. Note that some Azure services support local users and roles that are not managed through Azure AD. Customers will need to manage these users separately.

Responsibility: Customer

Microsoft Defender for Cloud monitoring: None

#### **PA-6: Use privileged access workstations**

Guidance: Azure Information Protection can be managed from a customer workstation through PowerShell.

Secured, isolated workstations are critically important for the security of sensitive roles, such as administrators, developers, and critical service operators.

Use highly secured user workstations and/or Azure Bastion for administrative tasks. Use Azure Active Directory (Azure AD), Microsoft Defender Advanced Threat Protection (ATP), and/or Microsoft Intune to deploy a secure and managed user workstation for administrative tasks. The secured workstations can be centrally managed to enforce secured configuration, including strong authentication, software and hardware baselines, and restricted logical and network access.

Responsibility: Customer

Microsoft Defender for Cloud monitoring: None

#### **PA-7: Follow just enough administration (least privilege principle)**

Guidance: Azure Information Protection is integrated with Azure Active Directory (Azure AD), which is Azure's default identity and access management service.

Azure Information Protection includes an administrator-level role in Azure AD. Users assigned to the Administrator role have full permissions in the Azure Information Protection service. Administrator role can be used to configure labels for the Azure Information Protection policy, managing protection templates, and activating protection. However, the Administrator role does not grant any permissions in Identity Protection Center, Privileged Identity Management, Monitor Microsoft 365 Service Health, or Office 365 Security & Compliance Center.

Limit the number of highly privileged accounts or roles and protect these accounts at an elevated level, as users with this privilege can directly or indirectly read and modify every resource in your Azure environment. Enable just-in-time (JIT) privileged access to Azure resources and Azure AD using Privileged Identity Management (PIM). Just-in-time access grants temporary permissions to perform privileged tasks only when users need it. PIM can also generate security alerts when there is suspicious or unsafe activity in your Azure AD organization.

Responsibility: Customer

Microsoft Defender for Cloud monitoring: None

#### **PA-8: Choose approval process for Microsoft support**

Guidance: Azure Information Protection supports Azure Customer Lockbox to provide customers with the ability to review, approve, and reject data access requests, as well as review requests being made.

Responsibility: Customer

Microsoft Defender for Cloud monitoring: None

### **Data Protection**

#### **DP-1: Discovery, classify and label sensitive data**

Guidance: Azure Information Protection provides the ability to discover, classify, and label sensitive information.

Azure Information Protection is a cloud-based solution that enables organizations to classify and protect documents and emails by applying labels. Labels can be applied automatically by administrators using rules and conditions, manually by users, or by a combination where administrators define the recommendations shown to users.

Responsibility: Shared

Microsoft Defender for Cloud monitoring: None

#### **DP-2: Protect sensitive data**

Guidance: Azure Information Protection provides data protection by offering the ability to label sensitive information and provide protection on that data through encryption. Protection is provided by the Azure Rights Management service.

Responsibility: Shared

Microsoft Defender for Cloud monitoring: None

#### **DP-3: Monitor for unauthorized transfer of sensitive data**

Guidance: Azure Information Protection provides the ability to monitor for unauthorized transfer of sensitive data through the track and revoke capability. Track and Revoke allows the customer to track how people are using documents they have sent and revoke access if people should no longer be able to read them.

Responsibility: Shared

Microsoft Defender for Cloud monitoring: None

## **Asset Management**

### **AM-1: Ensure security team has visibility into risks for assets**

Guidance: Ensure security teams are granted Security Reader permissions in your Azure tenant and subscriptions so they can monitor for security risks using Microsoft Defender for Cloud.

Depending on how security team responsibilities are structured, monitoring for security risks could be the responsibility of a central security team or a local team. That said, security insights and risks must always be aggregated centrally within an organization.

Security Reader permissions can be applied broadly to an entire tenant (Root Management Group) or scoped to management groups or specific subscriptions.

Note: Additional permissions might be required to get visibility into workloads and services.

Responsibility: Customer

Microsoft Defender for Cloud monitoring: None

### **AM-3: Use only approved Azure services**

Guidance: Azure Information Protection does not support Azure Resource Manager Deployments or allow customers the ability to limit deployments through built-in Azure Policy definitions, such as 'Allow Resources' or 'Deny Resources'. However, customers can limit usage of Azure Information Protection through labeling policies in the Security and Compliance Center.

Responsibility: Customer

Microsoft Defender for Cloud monitoring: None

## **Logging and Threat Detection**

### **LT-2: Enable threat detection for Azure identity and access management**

Guidance: Azure Information Protection is integrated with Azure Active Directory (Azure AD), which is Azure's default identity and access management service.

View Azure AD-provided user logs with Azure AD reporting and other solutions such as Azure Monitor, Microsoft Sentinel, or other SIEM/monitoring tools for more sophisticated monitoring and analytics use cases.

They are:

- Sign-in report – The sign-in report provides information about the usage of managed applications and user sign-in activities.
- Audit logs - Provides traceability through logs for all changes done by various features within Azure AD. Examples of audit logs include changes made to any

resources within Azure AD, such as adding or removing users, apps, groups, roles, and policies.

- Risky sign-ins - A risky sign-in is an indicator for a sign-in attempt that might have been performed by someone who is not the legitimate owner of a user account.
- Users flagged for risk - A risky user is an indicator for a user account that might have been compromised.

Microsoft Defender for Cloud can also alert on certain suspicious activities, such as an excessive number of failed authentication attempts, and deprecated accounts in the subscription. In addition to the basic security hygiene monitoring, Microsoft Defender for Cloud's Threat Protection module can also collect more in-depth security alerts from individual Azure compute resources (such as virtual machines, containers, app service), data resources (such as SQL DB and storage), and Azure service layers. This capability allows you to see account anomalies inside the individual resources.

Responsibility: Customer

Microsoft Defender for Cloud monitoring: None

#### **LT-4: Enable logging for Azure resources**

Guidance: Azure Information Protection provides data protection for an organization's documents and emails, along with a log for each request. These requests include when users protect documents and emails, when they consume this content, actions performed by administrators for this service, and actions performed by Microsoft operators to support your Azure Information Protection deployment.

Types of logs produced by Azure Information Protection include:

- Admin Log - Logs administrative tasks for the protection service. For example, if the service is deactivated, when the super user feature is enabled, and when users are delegated admin permissions to the service.
- Document Tracking - Lets users track and revoke their documents that they have tracked with the Azure Information Protection client. Global administrators can also track these documents on behalf of users.
- Client Event Logs - Usage activity for the Azure Information Protection client, logged in the local Windows Applications and Services event log, Azure Information Protection.
- Client Log Files - Troubleshooting logs for the Azure Information Protection client

The Protection usage logs can be used to identify 'who' is accessing your protected data, from 'which' devices, and from 'where'. Logs reveal whether people can successfully read protected content, as well as identify which people have read an important document that was protected.

Responsibility: Customer

Microsoft Defender for Cloud monitoring: None

## **LT-5: Centralize security log management and analysis**

Guidance: Ensure that support personnel can build a full view of what happened during an event, by querying and using diverse data sources, as they investigate potential incidents.

Avoid blind spots by collecting diverse logs and sending them to a central SIEM solution, such as Microsoft Sentinel, to track the activities of a potential attacker across the kill chain. The logs can reveal whether people can successfully read protected content, as well as identify which people have read an important document that was protected. Ensure that insights and learnings are captured for other analysts and for future historical reference.

Microsoft Sentinel provides extensive data analytics across virtually any log source and a case management portal to manage the full lifecycle of incidents. Intelligence information during an investigation can be associated with an incident for tracking and reporting purposes.

Responsibility: Customer

Microsoft Defender for Cloud monitoring: None

## **LT-6: Configure log storage retention**

Guidance: Azure Information Protection provides data protection for an organization's documents and emails, with a log for every request to it. These requests include when users protect documents and emails, when they consume this content, actions performed by your administrators for this service, and actions performed by Microsoft operators to support your Azure Information Protection deployment.

The amount of data collected and stored in your Azure Information Protection workspace, and its retention, will vary significantly for each tenant, depending on factors such as how many Azure Information Protection clients and other supported endpoints you have, whether you're collecting endpoint discovery data, you've deployed scanners, the number of protected documents that are accessed, and so on.

Use Azure Monitor Log's Usage and estimated costs feature to help estimate and review the amount of data stored and also control the data retention period for your Log Analytics workspace.

Responsibility: Customer

Microsoft Defender for Cloud monitoring: None

## **LT-7: Use approved time synchronization sources**

Guidance: Azure Information Protection does not support configuring your own time synchronization sources. The Azure Information Protection service relies on Microsoft time synchronization sources, and is not exposed to customers for configuration.

Responsibility: Microsoft

Microsoft Defender for Cloud monitoring: None

## Posture and Vulnerability Management

### PV-1: Establish secure configurations for Azure services

Guidance: Azure Information Protection can be configured through the Security and Compliance Center or through PowerShell.

Within the Security and Compliance Center, an admin can create sensitivity labels, define what each label can do, and publish the labels.

Create the labels: Create and name your sensitivity labels according to your organization's classification taxonomy for different sensitivity levels of content. Use common names or terms that make sense to your users. If you don't already have an established taxonomy, consider starting with label names such as Personal, Public, General, Confidential, and Highly Confidential. You can then use sublabels to group similar labels by category. When you create a label, use the tooltip text to help users select the appropriate label.

Define what each label can do: Configure the protection settings you want associated with each label. For example, you might want lower sensitivity content (such as a "General" label) to have just a header or footer applied, while higher sensitivity content (such as a "Confidential" label) should have a watermark and encryption.

Publish the labels: After your sensitivity labels are configured, publish them by using a label policy. Decide which users and groups should have the labels and what policy settings to use. A single label is reusable—you define it once, and then you can include it in several label policies assigned to different users. So for example, you could pilot your sensitivity labels by assigning a label policy to just a few users. Then when you're ready to roll out the labels across your organization, you can create a new label policy for your labels and this time, specify all users.

In order to use PowerShell, install the AIPService PowerShell Module. Within PowerShell, an admin can perform these tasks along with others:

- Migrate from on-premise Rights Management (AD RMS or Windows RMS) to Azure Information Protection
- Generate and Manage your own tenant key- the bring your own key (BYOK) scenario
- Activate or deactivate the Rights Management service for your organization
- Configure onboarding controls for a phased deployment of the Azure Rights Management service
- Create and manage Rights Management templates for your organization
- Manage users and groups who are authorized to administer Rights Management service for your organization
- Log and analyze usage for Rights Management

Responsibility: Customer

Microsoft Defender for Cloud monitoring: None

## **PV-8: Conduct regular attack simulation**

Guidance: As required, conduct penetration testing or red team activities on your Azure resources and ensure remediation of all critical security findings. Follow the Microsoft Cloud Penetration Testing Rules of Engagement to ensure your penetration tests are not in violation of Microsoft policies. Use Microsoft's strategy and execution of Red Teaming and live site penetration testing against Microsoft-managed cloud infrastructure, services, and applications.

Responsibility: Shared

Microsoft Defender for Cloud monitoring: None

## **Endpoint Security**

### **ES-1: Use Endpoint Detection and Response (EDR)**

Guidance: Not applicable; Azure Information Protection does not expose underlying infrastructure to customers. Customers cannot install any anti-malware solution through this offering.

Microsoft handles endpoint protection for the underlying infrastructure that supports the service.

Responsibility: Microsoft

Microsoft Defender for Cloud monitoring: None

### **ES-2: Use centrally managed modern anti-malware software**

Guidance: Not applicable; Azure Information Protection does not expose the underlying infrastructure to the customer. Customers cannot install any anti-malware solution through this offering.

Microsoft handles endpoint protection for the underlying infrastructure that supports the service.

Responsibility: Microsoft

Microsoft Defender for Cloud monitoring: None

### **ES-3: Ensure anti-malware software and signatures are updated**

Guidance: Not applicable; Azure Information Protection does not expose the underlying infrastructure to the customer. Customers cannot install any anti-malware solution through this offering.

Microsoft handles endpoint protection for the underlying infrastructure that supports the service.

Responsibility: Microsoft

Microsoft Defender for Cloud monitoring: None

## **Backup and Recovery**

### **BR-4: Mitigate risk of lost keys**

Guidance: Azure Information Protection provides customers with the ability to configure their tenant with their own key through Bring Your Own Key (BYOK). Customer-generated keys must be stored in Azure Key Vault for protection. Azure Key Vault helps prevent the loss of keys through soft delete, role separation, and separated security domains.

Responsibility: Customer

Microsoft Defender for Cloud monitoring: None

## **Azure Information Protection (AIP) labeling, classification, and protection**

Azure Information Protection (AIP) is a cloud-based solution that enables organizations to classify and protect documents and emails by applying labels.

For example, your administrator might configure a label with rules that detect sensitive data, such as credit card information. In this case, any user who saves credit card information in a Word file might see a tooltip at the top of the document with a recommendation to apply the relevant label for this scenario.

Labels can both classify, and optionally protect your documents, enabling you to:

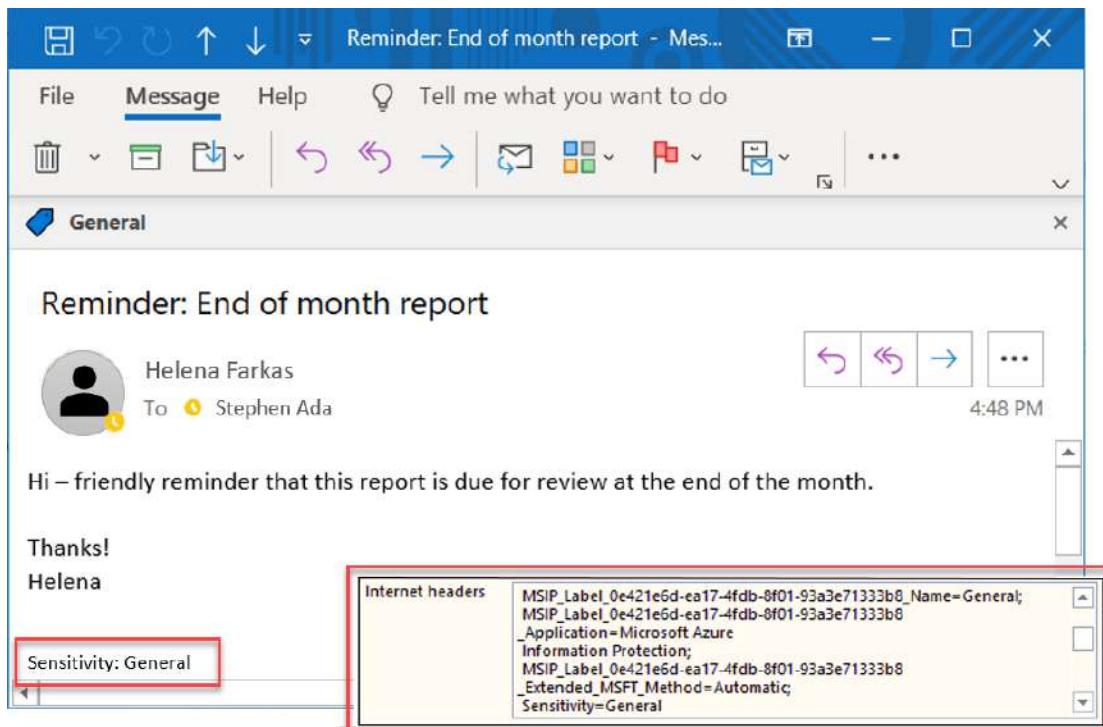
- Track and control how your content is used
- Analyze data flows to gain insight into your business - Detect risky behaviors and take corrective measures
- Track document access and prevent data leakage or misuse
- And more ...

## **How labels apply classification with AIP**

Labeling your content with AIP includes:

- Classification that can be detected regardless of where the data is stored or with whom it's shared.
- Visual markings, such as headers, footers, or watermarks.
- Metadata, added to files and email headers in clear text. The clear text metadata ensures that other services can identify the classification and take appropriate action

For example, in the image below, labeling has classified an email message as General:



In this example, the label also:

- Added a footer of Sensitivity: General to the email message. This footer is a visual indicator for all recipients that it's intended for general business data that should not be sent outside of the organization.
- Embedded metadata in the email headers. Header data enables email services to inspect the label and theoretically create an audit entry or prevent it from being sent outside of the organization.

Labels can be applied automatically by administrators using rules and conditions, manually by users, or using a combination where administrators define the recommendations shown to users.

## How AIP protects your data

Azure Information Protection uses the Azure Rights Management service (Azure RMS) to protect your data.

Azure RMS is integrated with other Microsoft cloud services and applications, such as Office 365 and Azure Active Directory, and can also be used with your own or third-party applications and information protection solutions. Azure RMS works with both on-premises and cloud solutions.

Azure RMS uses encryption, identity, and authorization policies. Similar to AIP labels, protection applied using Azure RMS stays with the documents and emails, regardless of the document or email's location, ensuring that you stay in control of your content even when it's shared with other people.

Protection settings can be:

- Part of your label configuration, so that users both classify and protect documents and emails simply by applying a label.

- Used on their own, by applications and services that support protection but not labeling.

For applications and services that support protection only, protection settings are used as Rights Management templates.

For example, you may want to configure a report or sales forecast spreadsheet so that it can be accessed only by people in your organization. In this case, you'd apply protection settings to control whether that document can be edited, restrict it to read-only, or prevent it from being printed.

Emails can have similar protection settings to prevent them from being forwarded or from using the Reply All option.

## Rights Management templates

As soon as the Azure Rights Management service is activated, two default rights management templates are available for you to restrict data access to users within your organization. Use these templates immediately, or configure your own protection settings to apply more restrictive controls in new templates.

Rights Management templates can be used with any applications or services that support Azure Rights Management.

The following image shows an example from the Exchange admin center, where you can configure Exchange Online mail flow rules to use RMS templates:

Name:

\*Apply this rule if...  ['New Launch Team'](#)

\*Do the following...

Apply rights protection to the message

Except if...

select RMS template

RMS template:

- Sales and Marketing - Read and Print Only
- Sales and Marketing - Read and Print Only** (VanArsdel, Ltd - Confidential View Only)
- VanArsdel, Ltd - Confidential
- Do Not Forward

### Note

Creating an AIP label that includes protection settings also creates a corresponding Rights Management template that can be used separately from the label.

## AIP and end-user integration for documents and emails

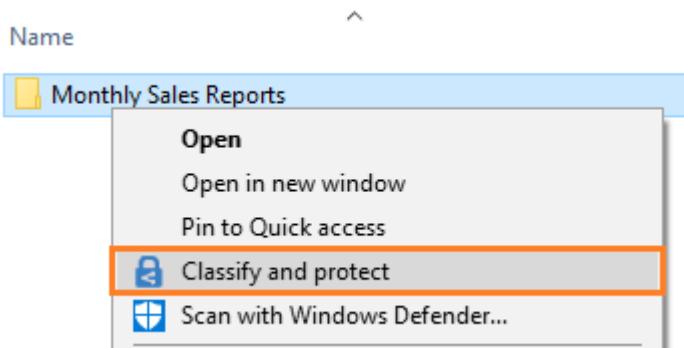
The AIP client installs the Information Protection bar to Office applications and enables end users to integrate AIP with their documents and emails.

For example, in Excel:

For Reorder	Inventory ID	Name	Description	Unit Price	Quantity in Stock	Inventory Value	Reorder Level	Reorder Time in Days
	IN0001	Item 1	Desc 1	\$51.00	25	\$1,275.00	29	13
	IN0002	Item 2	Desc 2	\$93.00	132	\$12,276.00	231	4
	IN0003	Item 3	Desc 3	\$57.00	151	\$8,607.00	114	11
	IN0004	Item 4	Desc 4	\$19.00	186	\$3,534.00	158	6

While labels can be applied automatically to documents and emails, removing guesswork for users or to comply with an organization's policies, the Information Protection bar enables end users to select labels and apply classification on their own.

Additionally, the AIP client enables users to classify and protect additional file types, or multiple files at once, using the right-click menu from Windows File Explorer. For example:

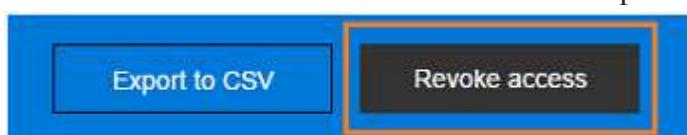


The Classify and protect menu option works similarly to the Information Protection bar in Office applications, enabling users to select a label or set custom permissions.

Tip

Power users or administrators might find that PowerShell commands are more efficient for managing and setting classification and protection for multiple files. Relevant PowerShell commands are included with the client, and can also be installed separately.

Users and administrators can use document tracking sites to monitor protected documents, watch who accesses them, and when. If they suspect misuse, they can also revoke access to these documents. For example:



## Additional integration for email

Using AIP with Exchange Online provides the additional benefit of sending protected emails to any user, with the assurance that they can read it on any device.

For example, you may need to send sensitive information to personal email addresses that use a Gmail, Hotmail, or Microsoft account, or to users who don't have an account in Office 365 or Azure AD. These emails should be encrypted at rest and in transit, and be read only by the original recipients.

This scenario requires Office 365 Message Encryption capabilities. If the recipients cannot open the protected email in their built-in email client, they can use a one-time passcode to read the sensitive information in a browser.

For example, a Gmail user might see the following prompt in an email message they receive:

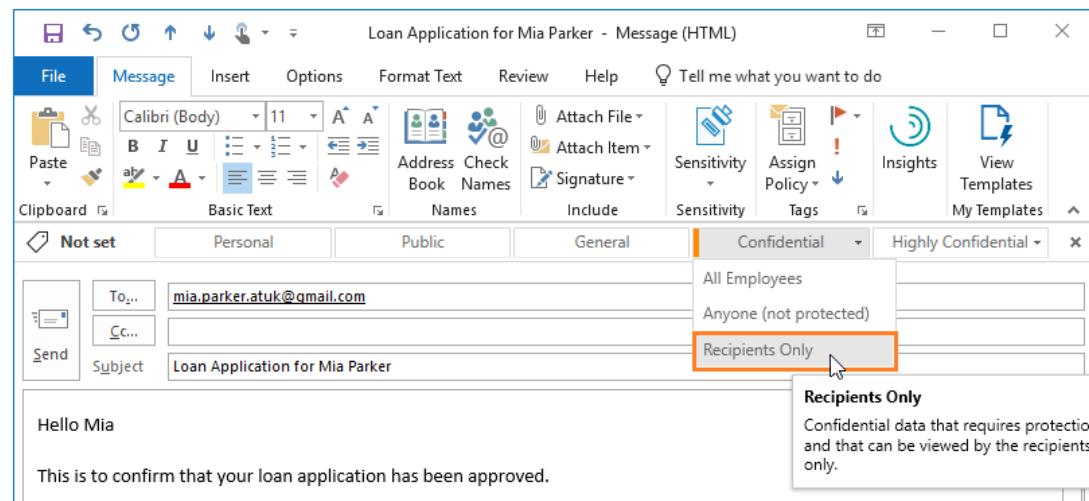
Sign in to view the message



Or, sign in with a one-time passcode

For the user sending the email, the actions required are the same as for sending a protected email to a user in their own organization. For example, select the Do Not Forward button that the AIP client can add to the Outlook ribbon.

Alternately, Do Not Forward functionality can be integrated into a label that users can select to apply both classification and protection to that email. For example:



Administrators can also automatically provide protection for users by configuring mail flow rules that apply rights protection.

Any Office documents attached to these emails are automatically protected as well.

## Scanning for existing content to classify and protect

Ideally, you'll be labeling documents and emails as they're created. However, you likely have many existing documents, stored either on-premises or in the cloud, and want to classify and protect these documents as well.

Use one of the following methods to classify and protect existing content:

- On-premises storage: Use the Azure Information Protection scanner to discover, classify, and protect documents on network shares and Microsoft SharePoint Server sites and libraries.

The scanner runs as a service on Windows Server, and uses the same policy rules to detect sensitive information and apply specific labels to documents.

Alternatively, use the scanner to apply a default label to all documents in a data repository without inspecting the file contents. Use the scanner in reporting mode only to discover sensitive information that you might not know you had.

- Cloud data storage: Use Microsoft Defender for Cloud Apps to apply your labels to documents in Box, SharePoint, and OneDrive.

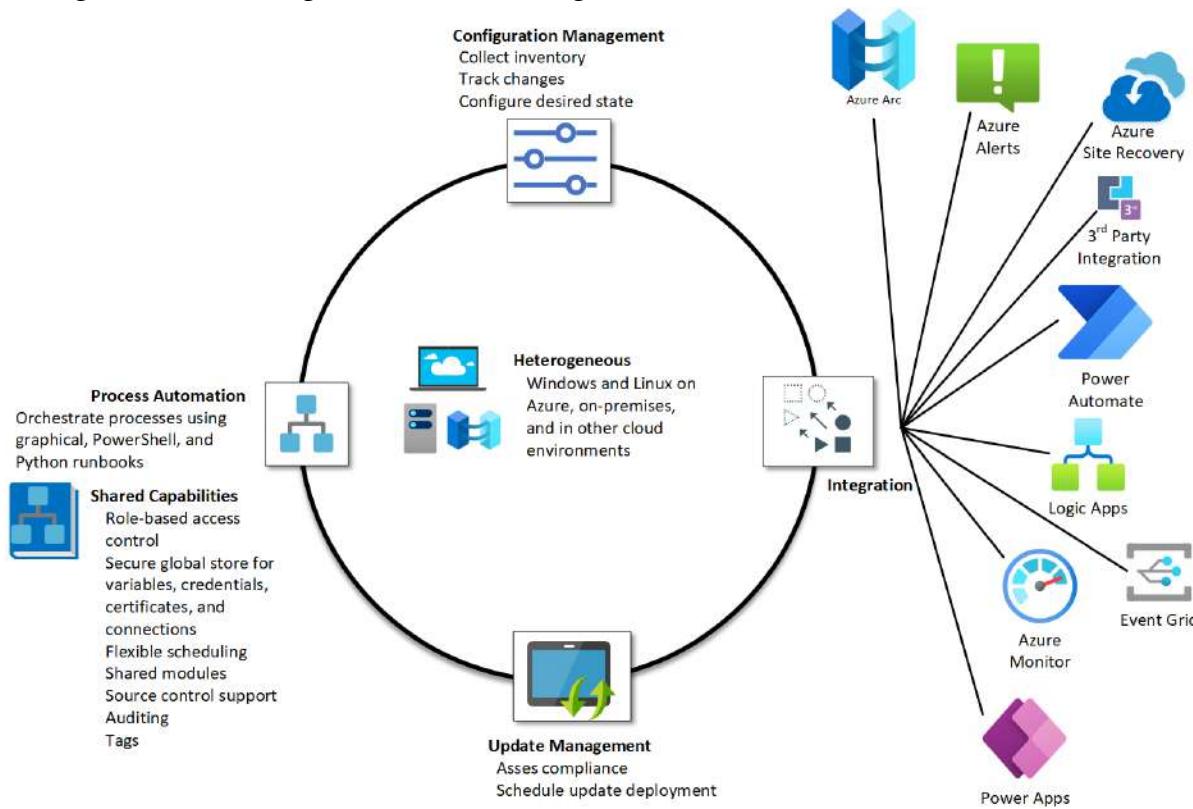
# AZURE AUTOMATION

## What is Azure Automation?

Automation is needed in three broad areas of cloud operations:

- Deploy and manage - Deliver repeatable and consistent infrastructure as code.
- Response - Create event-based automation to diagnose and resolve issues.
- Orchestrate - Orchestrate and integrate your automation with other Azure or third party services and products.

Azure Automation delivers a cloud-based automation, operating system updates, and configuration service that supports consistent management across your Azure and non-Azure environments. It includes process automation, configuration management, update management, shared capabilities, and heterogeneous features.



There are several Azure services that can deliver the above requirements, where each service includes a set of capabilities and serves a role as a programmable platform to build cloud solutions. For example, Azure Bicep and Resource Manager provide a language to develop repeatable and consistent deployment templates for Azure resources. Azure Automation can process that template to deploy an Azure resource and then process a set of post-deployment configuration tasks.

Automation gives you complete control during deployment, operations, and decommissioning of enterprise workloads and resources.

## **Process Automation**

Process Automation in Azure Automation allows you to automate frequent, time-consuming, and error-prone management tasks. This service helps you focus on work that adds business value. By reducing errors and boosting efficiency, it also helps to lower your operational costs.

Process automation supports the integration of Azure services and other third party systems required in deploying, configuring, and managing your end-to-end processes. The service allows you to author graphical, PowerShell and Python runbooks. To run runbooks directly on the Windows or Linux machine or against resources in the on-premises or other cloud environment to manage those local resources, you can deploy a Hybrid Runbook Worker to the machine.

Webhooks let you fulfil requests and ensure continuous delivery and operations by triggering automation from Azure Logic Apps, Azure Function, ITSM product or service, DevOps, and monitoring systems.

## **Configuration Management**

Configuration Management in Azure Automation is supported by two capabilities:

- Change Tracking and Inventory
- Azure Automation State Configuration

### **Change Tracking and Inventory**

Change Tracking and Inventory combines functions to allow you to track Linux and Windows virtual machine and server infrastructure changes. The service supports change tracking across services, daemons, software, registry, and files in your environment to help you diagnose unwanted changes and raise alerts. Inventory support allows you to query in-guest resources for visibility into installed applications and other configuration items.

### **Azure Automation State Configuration**

Azure Automation State Configuration is a cloud-based feature for PowerShell desired state configuration (DSC) that provides services for enterprise environments. Using this feature, you can manage your DSC resources in Azure Automation and apply configurations to virtual or physical machines from a DSC pull server in the Azure cloud.

## **Update Management**

Azure Automation includes the Update Management feature for Windows and Linux systems across hybrid environments. Update Management gives you visibility into update compliance across Azure and other clouds, and on-premises. The feature allows you to create scheduled deployments that orchestrate the installation of updates within a defined

maintenance window. If an update shouldn't be installed on a machine, you can use Update Management functionality to exclude it from a deployment.

## Shared capabilities

Azure Automation provides a number of shared capabilities, including shared resources, role-based access control, flexible scheduling, source control integration, auditing, and tagging.

### Shared resources

Azure Automation consists of a set of shared resources that make it easier to automate and configure your environments at scale.

- Schedules - Trigger Automation operations at predefined times.
- Modules - Manage Azure and other systems. You can import modules into the Automation account for Microsoft, third-party, community, and custom-defined cmdlets and DSC resources.
- Modules gallery - Supports native integration with the PowerShell Gallery to let you view runbooks and import them into the Automation account. The gallery allows you to quickly get started integrating and authoring your processes from PowerShell gallery and Microsoft Script Center.
- Python 2 and 3 packages - Support Python 2 and 3 runbooks for your Automation account.
- Credentials - Securely store sensitive information that runbooks and configurations can use at runtime.
- Connections - Store name-value pairs of common information for connections to systems. The module author defines connections in runbooks and configurations for use at runtime.
- Certificates - Define information to be used in authentication and securing of deployed resources when accessed by runbooks or DSC configurations at runtime.
- Variables - Hold content that can be used across runbooks and configurations. You can change variable values without having to modify any of the runbooks or configurations that reference them.

### Role-based access control

Azure Automation supports Azure role-based access control (Azure RBAC) to regulate access to the Automation account and its resources.

### Source control integration

Azure Automation supports source control integration. This feature promotes configuration as code where runbooks or configurations can be checked into a source control system.

## Heterogeneous support (Windows and Linux)

Automation is designed to work across Windows and Linux physical servers and virtual machines outside of Azure, on your corporate network, or other cloud provider. It delivers a consistent way to automate and configure deployed workloads and the operating systems that run them. The Hybrid Runbook Worker feature of Azure Automation enables running runbooks directly on the non-Azure physical server or virtual machine hosting the role, and against resources in the environment to manage those local resources.

Through Arc-enabled servers, it provides a consistent deployment and management experience for your non-Azure machines. It enables integration with the Automation service using the VM extension framework to deploy the Hybrid Runbook Worker role, and simplify onboarding to Update Management and Change Tracking and Inventory.

## Common scenarios

Azure Automation supports management throughout the lifecycle of your infrastructure and applications. Common scenarios include:

- Schedule tasks - stop VMs or services at night and turn on during the day, weekly or monthly recurring maintenance workflows.
- Write runbooks - Author PowerShell, PowerShell Workflow, graphical, Python 2 and 3, and DSC runbooks in common languages.
- Build and deploy resources - Deploy virtual machines across a hybrid environment using runbooks and Azure Resource Manager templates. Integrate into development tools, such as Jenkins and Azure DevOps.
- Configure VMs - Assess and configure Windows and Linux machines with configurations for the infrastructure and application.
- Share knowledge - Transfer knowledge into the system on how your organization delivers and maintains workloads.
- Retrieve inventory - Get a complete inventory of deployed resources for targeting, reporting, and compliance.
- Find changes - Identify and isolate machine changes that can cause misconfiguration and improve operational compliance. Remediate or escalate them to management systems.
- Periodic maintenance - to execute tasks that need to be performed at set timed intervals like purging stale or old data, or reindexing a SQL database.
- Respond to alerts - Orchestrate a response when cost-based, system-based, service-based, and/or resource utilisation alerts are generated.
- Hybrid automation - Manage or automate on-premises servers and services like SQL Server, Active Directory, SharePoint Server, etc.
- Azure resource lifecycle management - for IaaS and PaaS services.
- Dev/test automation scenarios - Start and start resources, scale resources, etc.
- Governance related automation - Automatically apply or update tags, locks, etc.

- Azure Site Recovery - orchestrate pre/post scripts defined in a Site Recovery DR workflow.
- Windows Virtual Desktop - orchestrate scaling of VMs or start/stop VMs based on utilization.

Depending on your requirements, one or more of the following Azure services integrate with or compliment Azure Automation to help fulfil them:

- Azure Arc-enabled servers enable simplified onboarding of hybrid machines to Update Management, Change Tracking and Inventory, and the Hybrid Runbook Worker role.
- Azure Alerts action groups can initiate an Automation runbook when an alert is raised.
- Azure Monitor to collect metrics and log data from your Automation account for further analysis and take action on the telemetry. Automation features such as Update Management and Change Tracking and Inventory rely on the Log Analytics workspace to deliver elements of their functionality.
- Azure Policy includes initiative definitions to help establish and maintain compliance with different security standards for your Automation account.
- Azure Site Recovery can use Azure Automation runbooks to automate recovery plans.

These Azure services can work with Automation job and runbook resources using an HTTP webhook or API method:

- Azure Logic Apps
- Azure Power Apps
- Azure Event Grid
- Azure Power Automate

## Azure Automation account authentication

### Automation account

When you start Azure Automation for the first time, you must create at least one Automation account. Automation accounts allow you to isolate your Automation resources, runbooks, assets, and configurations from the resources of other accounts. You can use Automation accounts to separate resources into separate logical environments or delegated responsibilities. For example, you might use one account for development, another for production, and another for your on-premises environment. Or you might dedicate an Automation account to manage operating system updates across all of your machines with Update Management.

An Azure Automation account is different from your Microsoft account or accounts created in your Azure subscription.

## Automation resources

The Automation resources for each Automation account are associated with a single Azure region, but the account can manage all the resources in your Azure subscription. The main reason to create Automation accounts in different regions is if you have policies that require data and resources to be isolated to a specific region.

All tasks that you create against resources using Azure Resource Manager and the PowerShell cmdlets in Azure Automation must authenticate to Azure using Azure Active Directory (Azure AD) organizational identity credential-based authentication.

## Managed identities

A managed identity from Azure Active Directory (Azure AD) allows your runbook to easily access other Azure AD-protected resources. The identity is managed by the Azure platform and doesn't require you to provision or rotate any secrets.

Managed identities are the recommended way to authenticate in your runbooks, and is the default authentication method for your Automation account.

Note: When you create an Automation account, the option to create a Run As account is no longer available. However, we continue to support a RunAs account for existing and new Automation accounts. You can create a Run As account in your Automation account from the Azure portal or by using PowerShell.

Here are some of the benefits of using managed identities:

- Using a managed identity instead of the Automation Run As account simplifies management. You don't have to renew the certificate used by a Run As account.
- Managed identities can be used without any additional cost.
- You don't have to specify the Run As connection object in your runbook code. You can access resources using your Automation account's managed identity from a runbook without creating certificates, connections, Run As accounts, etc.

An Automation account can authenticate using two types of managed identities:

- A system-assigned identity is tied to your application and is deleted if your app is deleted. An app can only have one system-assigned identity.
- A user-assigned identity is a standalone Azure resource that can be assigned to your app. An app can have multiple user-assigned identities.

## Run As accounts

Run As accounts in Azure Automation provide authentication for managing Azure Resource Manager resources or resources deployed on the classic deployment model. There are two types of Run As accounts in Azure Automation:

To create or renew a Run As account, permissions are needed at three levels:

- Subscription,
- Azure Active Directory (Azure AD), and
- Automation account

## Subscription permissions

You need the Microsoft.Authorization/\*/Write permission. This permission is obtained through membership of one of the following Azure built-in roles:

- Owner
- User Access Administrator

To configure or renew Classic Run As accounts, you must have the Co-administrator role at the subscription level.

## Azure AD permissions

To be able to create or renew the service principal, you need to be a member of one of the following Azure AD built-in roles:

- Application Administrator
- Application Developer

Membership can be assigned to ALL users in the tenant at the directory level, which is the default behaviour. You can grant membership to either role at the directory level.

## Automation account permissions

To be able to create or update the Automation account, you need to be a member of one of the following Automation account roles:

- Owner
- Contributor
- Custom Azure Automation Contributor

Note: Azure Cloud Solution Provider (CSP) subscriptions support only the Azure Resource Manager model. Non-Azure Resource Manager services are not available in the program. When you are using a CSP subscription, the Azure Classic Run As account is not created, but the Azure Run As account is created.

When you create an Automation account, the Run As account is created by default at the same time with a self-signed certificate. If you chose not to create it along with the Automation account, it can be created individually at a later time. An Azure Classic Run As Account is optional, and is created separately if you need to manage classic resources.

Note: Azure Automation does not automatically create the Run As account. It has been replaced by using managed identities.

If you want to use a certificate issued by your enterprise or third-party certification authority (CA) instead of the default self-signed certificate, you can use the PowerShell script to create a Run As account option for your Run As and Classic Run As accounts.

## Run As account

When you create a Run As account, it performs the following tasks:

- Creates an Azure AD application with a self-signed certificate, creates a service principal account for the application in Azure AD, and assigns the Contributor role for the account in your current subscription. You can change the certificate setting to Reader or any other role.
- Creates an Automation certificate asset named AzureRunAsCertificate in the specified Automation account. The certificate asset holds the certificate private key that the Azure AD application uses.
- Creates an Automation connection asset named AzureRunAsConnection in the specified Automation account. The connection asset holds the application ID, tenant ID, subscription ID, and certificate thumbprint.

## Azure Classic Run As account

When you create an Azure Classic Run As account, it performs the following tasks:

Note: You must be a co-administrator on the subscription to create or renew this type of Run As account.

- Creates a management certificate in the subscription.
- Creates an Automation certificate asset named AzureClassicRunAsCertificate in the specified Automation account. The certificate asset holds the certificate private key used by the management certificate.
- Creates an Automation connection asset named AzureClassicRunAsConnection in the specified Automation account. The connection asset holds the subscription name, subscription ID, and certificate asset name.

## Service principal for Run As account

The service principal for a Run As account doesn't have permissions to read Azure AD by default. If you want to add permissions to read or manage Azure AD, you must grant the permissions on the service principal under API permissions.

## Run As account permissions

This section defines permissions for both regular Run As accounts and Classic Run As accounts.

- To create or update a Run As account, an Application administrator in Azure Active Directory and an Owner in the subscription can complete all the tasks.

- To configure or renew Classic Run As accounts, you must have the Co-administrator role at the subscription level.

In a situation where you have separation of duties, the following table shows a listing of the tasks, the equivalent cmdlet, and permissions needed:

Task	Cmdlet	Minimum Permissions	Where you set the permissions
Create Azure AD Application	New-AzADApplication	Application Developer role1	Azure AD Home > Azure AD > App Registrations
Add a credential to the application.	New-AzADAppCredential	Application Administrator or Global Administrator1	Azure AD Home > Azure AD > App Registrations
Create and get an Azure AD service principal	New-AzADServicePrincipal   Get-AzADServicePrincipal	Application Administrator or Global Administrator1	Azure AD Home > Azure AD > App Registrations

Assign or get the Azure role for the specified principal	New-AzRoleAssignment Get-AzRoleAssignment	User Access Administrator or Owner Subscription or have the following permissions: Home > Subscription Microsoft.Authorization/Operations/ read Microsoft.Authorization/permissions - read Microsoft.Authorization/roleDefinitions/read Microsoft.Authorization/roleAssignments/write Microsoft.Authorization/roleAssignments/read Microsoft.Authorization/roleAssignments/delete	
Create or remove an Automation certificate	New-AzAutomationCertificate Remove-AzAutomationCertificate	Contributor on resource group	Automation account resource group
Create or remove an Automation connection	New-AzAutomationConnection Remove-AzAutomationConnection	Contributor on resource group	Automation account resource group

1 Non-administrator users in your Azure AD tenant can register AD applications if the Azure AD tenant's Users can register applications option on the User settings page is set to Yes. If the application registration setting is No, the user performing this action must be as defined in this table.

If you aren't a member of the subscription's Active Directory instance before you're added to the Global Administrator role of the subscription, you're added as a guest. In this situation, you receive a You do not have permissions to create... warning on the Add Automation account page.

To verify that the situation producing the error message has been remedied:

1. From the Azure Active Directory pane in the Azure portal, select Users and groups.
2. Select All users.
3. Choose your name, then select Profile.
4. Ensure that the value of the User type attribute under your user's profile isn't set to Guest.

## Role-based access control

Role-based access control is available with Azure Resource Manager to grant permitted actions to an Azure AD user account and Run As account, and authenticate the service principal.

If you have strict security controls for permission assignment in resource groups, you need to assign the Run As account membership to the Contributor role in the resource group.

Note: We recommend you don't use the Log Analytics Contributor role to execute Automation jobs. Instead, create the Azure Automation Contributor custom role and use it for actions related to the Automation account.

## Runbook authentication with Hybrid Runbook Worker

Runbooks running on a Hybrid Runbook Worker in your datacenter or against computing services in other cloud environments like AWS, can't use the same method that is typically used for runbooks authenticating to Azure resources. This is because those resources are running outside of Azure and therefore, require their own security credentials defined in Automation to authenticate to resources that they access locally.

For runbooks that use Hybrid Runbook Workers on Azure VMs, you can use runbook authentication with managed identities instead of Run As accounts to authenticate to your Azure resources.

## Runbook execution in Azure Automation

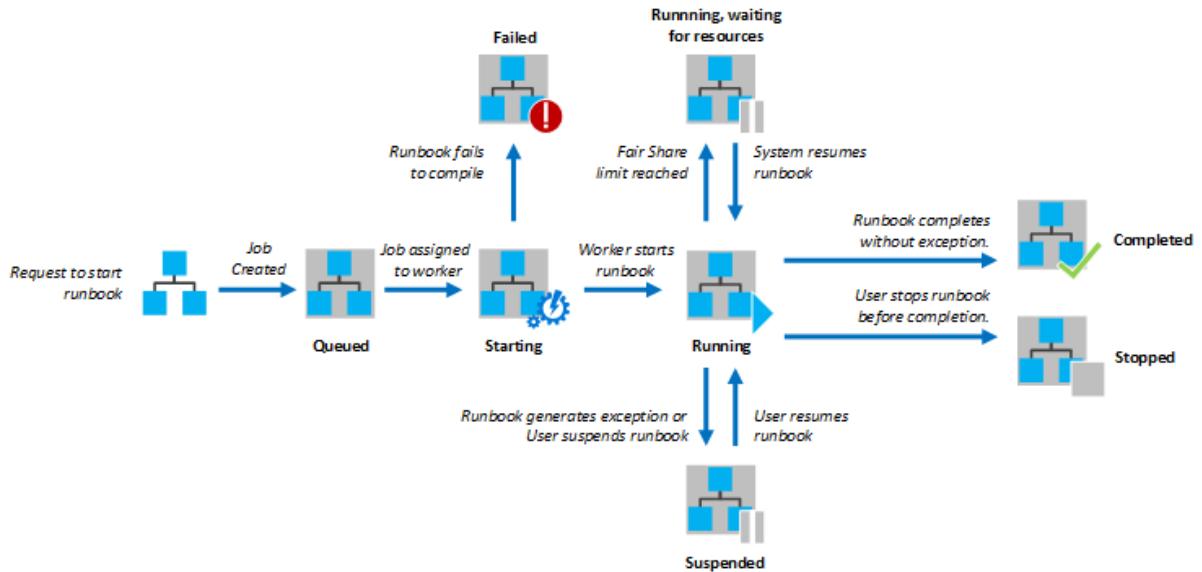
Automation executes your runbooks based on the logic defined inside them. If a runbook is interrupted, it restarts at the beginning. This behaviour requires you to write runbooks that support being restarted if transient issues occur.

Starting a runbook in Azure Automation creates a job, which is a single execution instance of the runbook. Each job accesses Azure resources by making a connection to your Azure subscription. The job can only access resources in your datacenter if those resources are accessible from the public cloud.

Azure Automation assigns a worker to run each job during runbook execution. While workers are shared by many Automation accounts, jobs from different Automation accounts are isolated from one another. You can't control which worker services your job requests.

When you view the list of runbooks in the Azure portal, it shows the status of each job that has been started for each runbook. Azure Automation stores job logs for a maximum of 30 days.

The following diagram shows the lifecycle of a runbook job for PowerShell runbooks, PowerShell Workflow runbooks, and graphical runbooks.



## Runbook execution environment

Runbooks in Azure Automation can run on either an Azure sandbox or a Hybrid Runbook Worker. When runbooks are designed to authenticate and run against resources in Azure, they run in an Azure sandbox. Azure Automation assigns a worker to run each job during runbook execution in the sandbox. While workers are shared by many Automation accounts, jobs from different Automation accounts are isolated from one another. Jobs using the same sandbox are bound by the resource limitations of the sandbox. The Azure sandbox environment does not support interactive operations. It prevents access to all out-of-process COM servers, and it does not support making WMI calls to the Win32 provider in your runbook. These scenarios are only supported by running the runbook on a Windows Hybrid Runbook Worker.

You can also use a Hybrid Runbook Worker to run runbooks directly on the computer that hosts the role and against local resources in the environment. Azure Automation stores and manages runbooks and then delivers them to one or more assigned computers.

Enabling the Azure Firewall on Azure Storage, Azure Key Vault, or Azure SQL blocks access from Azure Automation runbooks for those services. Access will be blocked even when the firewall exception to allow trusted Microsoft services is enabled, as Automation is not a part of the trusted services list. With an enabled firewall, access can only be made by using a Hybrid Runbook Worker and a virtual network service endpoint.

**Note:** To run on a Linux Hybrid Runbook Worker, your scripts must be signed and the worker configured accordingly. Alternatively, signature validation must be turned off.

The following table lists some runbook execution tasks with the recommended execution environment listed for each.

Task	Recommendation	Notes
Integrate with Azure resources	Azure Sandbox	Hosted in Azure, authentication is simple. If you're using a Hybrid Runbook Worker on an Azure VM, you can use runbook authentication with managed identities.
Obtain optimal performance to manage Azure resources	Azure Sandbox	Script is run in the same environment, which has less latency.
Minimize operational costs	Azure Sandbox	There is no compute overhead and no need for a VM.
Execute long-running script	Hybrid Runbook Worker	Azure sandboxes have resource limits.
Interact with local service	Hybrid Runbook Worker	Directly access the host machine, or resources in other cloud environments from the on-premises environment.
Require third-party software and executable	Hybrid Runbook Worker	You manage the operating system and can install software.
Monitor a file or folder with a runbook	Hybrid Runbook Worker	Use a Watcher task on a Hybrid Runbook Worker.
Run a resource-intensive script	Hybrid Runbook Worker	Azure sandboxes have resource limits.

---

Use modules with specific requirements	Hybrid Runbook Worker	Some examples are: WinSCP - dependency on winscp.exe IIS administration - dependency on enabling or managing IIS
Install a module with an installer	Hybrid Runbook Worker	Modules for sandbox must support copying.
Use runbooks or modules that require .NET Framework version different from 4.7.2	Hybrid Runbook Worker	Azure sandboxes support .NET Framework 4.7.2, and upgrading to a different version is not supported.
Run scripts that require elevation	Hybrid Runbook Worker	Sandboxes don't allow elevation. With Hybrid Runbook Worker, you can turn off UAC and use Invoke-Command when running the command that requires elevation.
Run scripts that require access to Windows Management Instrumentation (WMI)	Hybrid Runbook Worker	Jobs running in sandboxes in the cloud can't access WMI providers.

## Temporary storage in a sandbox

If you need to create temporary files as part of your runbook logic, you can use the Temp folder (that is, \$env:TEMP) in the Azure sandbox for runbooks running in Azure. The only limitation is you cannot use more than 1 GB of disk space, which is the quota for each sandbox. When working with PowerShell workflows, this scenario can cause a problem because PowerShell workflows use checkpoints and the script could be retried in a different sandbox.

With the hybrid sandbox, you can use C:\temp based on the availability of storage on a Hybrid Runbook Worker. However, per Azure VM recommendations, you should not use the temporary disk on Windows or Linux for data that needs to be persisted.

## Resources

Your runbooks must include logic to deal with resources, for example, VMs, the network, and resources on the network. Resources are tied to an Azure subscription, and runbooks require appropriate credentials to access any resource.

## Security

Azure Automation uses the Microsoft Defender for Cloud to provide security for your resources and detect compromise in Linux systems. Security is provided across your workloads, whether resources are in Azure or not.

Defender for Cloud places constraints on users who can run any scripts, either signed or unsigned, on a VM. If you are a user with root access to a VM, you must explicitly configure the machine with a digital signature or turn it off. Otherwise, you can only run a script to apply operating system updates after creating an Automation account and enabling the appropriate feature.

## Subscriptions

An Azure subscription is an agreement with Microsoft to use one or more cloud-based services, for which you are charged. For Azure Automation, each subscription is linked to an Azure Automation account, and you can create multiple subscriptions in the account.

## Credentials

A runbook requires appropriate credentials to access any resource, whether for Azure or third-party systems. These credentials are stored in Azure Automation, Key Vault, etc.

## Azure Monitor

Azure Automation makes use of Azure Monitor for monitoring its machine operations. The operations require a Log Analytics workspace and a Log Analytics agent.

### Log Analytics agent for Windows

The Log Analytics agent for Windows works with Azure Monitor to manage Windows VMs and physical computers. The machines can be running either in Azure or in a non-Azure environment, such as a local datacenter.

### Log Analytics agent for Linux

The Log Analytics agent for Linux works similarly to the agent for Windows, but connects Linux computers to Azure Monitor. The agent is installed with certain service accounts that execute commands requiring root permissions.

The Log Analytics agent log is located at /var/opt/microsoft/omsagent/log/omsagent.log.

## Runbook permissions

A runbook needs permissions for authentication to Azure, through credentials.

## Modules

Azure Automation includes the following PowerShell modules:

- Orchestrator.AssetManagement.Cmdlets - contains several internal cmdlets that are only available when you execute runbooks in the Azure sandbox environment or on a Windows Hybrid Runbook Worker. These cmdlets are designed to be used instead of Azure PowerShell cmdlets to interact with your Automation account resources.
- Az.Automation - the recommended PowerShell module for interacting with Azure Automation that replaces the AzureRM Automation module. The Az.Automation module is not automatically included when you create an Automation account and you need to import them manually.
- AzureRM.Automation - installed by default when you create an Automation account.

Also supported are installable modules, based on the cmdlets that your runbooks and DSC configurations require.

## Certificates

Azure Automation uses certificates for authentication to Azure or adds them to Azure or third-party resources. The certificates are stored securely for access by runbooks and DSC configurations.

Your runbooks can use self-signed certificates, which are not signed by a certificate authority (CA).

## Jobs

Azure Automation supports an environment to run jobs from the same Automation account. A single runbook can have many jobs running at one time. The more jobs you run at the same time, the more often they can be dispatched to the same sandbox.

Jobs running in the same sandbox process can affect each other. One example is running the Disconnect-AzAccount cmdlet. Execution of this cmdlet disconnects each runbook job in the shared sandbox process.

## Job statuses

The following table describes the statuses that are possible for a job. You can view a status summary for all runbook jobs or drill into details of a specific runbook job in the Azure portal. You can also configure integration with your Log Analytics workspace to forward runbook job status and job streams.

Status	Description
Activating	The job is being activated.
Completed	The job was completed successfully.
Failed	A graphical or PowerShell Workflow runbook failed to compile. A PowerShell runbook failed to start or the job had an exception.
Failed, waiting for resources	The job failed because it reached the fair share limit three times and started from the same checkpoint or from the start of the runbook each time.
Queued	The job is waiting for resources on an Automation worker to become available so that it can be started.
Resuming	The system is resuming the job after it was suspended.
Running	The job is running.
Running, waiting for resources	The job has been unloaded because it reached the fair share limit. It will resume shortly from its last checkpoint.
Starting	The job has been assigned to a worker, and the system is starting it.

---

Stopped	The job was stopped by the user before it was completed.
Stopping	The system is stopping the job.
Suspended	Applies to graphical and PowerShell Workflow runbooks only. The job is suspended by the user, by the system, or by a command in the runbook. If a runbook doesn't have a checkpoint, it starts from the beginning. If it has a checkpoint, it can start again and resume from its last checkpoint. The system only suspends the runbook when an exception occurs. By default, the ErrorActionPreference variable is set to Continue, indicating that the job keeps running on an error. If the preference variable is set to Stop, the job suspends on an error.
Suspending	Applies to graphical and PowerShell Workflow runbooks only. The system is trying to suspend the job at the request of the user. The runbook needs to reach its next checkpoint before it can be suspended. If it has already passed its last checkpoint, it completes before it can be suspended.

## Activity logging

Execution of runbooks in Azure Automation writes details in an activity log for the Automation account.

## Exceptions

This section describes some ways to handle exceptions or intermittent issues in your runbooks. An example is a WebSocket exception. Correct exception handling prevents transient network failures from causing your runbooks to fail.

### ErrorActionPreference

The `ErrorActionPreference` variable determines how PowerShell responds to a non-terminating error. Terminating errors always terminate and are not affected by `ErrorActionPreference`.

When the runbook uses `ErrorActionPreference`, a normally non-terminating error such as `PathNotFound` from the `Get-ChildItem` cmdlet stops the runbook from completing. The following example shows the use of `ErrorActionPreference`. The final `Write-Output` command never executes, as the script stops.

```
$ErrorActionPreference = 'Stop'  
Get-ChildItem -path nofile.txt  
Write-Output "This message will not show"
```

## Try Catch Finally

[Try Catch Finally](#) is used in PowerShell scripts to handle terminating errors. The script can use this mechanism to catch specific exceptions or general exceptions. The catch statement should be used to track or try to handle errors. The following example tries to download a file that does not exist. It catches the System.Net.WebException exception and returns the last value for any other exception.

```
try  
{  
    $wc = new-object System.Net.WebClient  
    $wc.DownloadFile("http://www.contoso.com/MyDoc.doc")  
}  
catch [System.Net.WebException]  
{  
    "Unable to download MyDoc.doc from http://www.contoso.com."  
}  
catch  
{  
    "An error occurred that could not be resolved."  
}
```

## Throw

[Throw](#) can be used to generate a terminating error. This mechanism can be useful when defining your own logic in a runbook. If the script meets a criterion that should stop it, it can use the throw statement to stop. The following example uses this statement to show a required function parameter.

```
function Get-ContosoFiles  
{  
    param ($path = $(throw "The Path parameter is required."))  
    Get-ChildItem -Path $path\*.txt -recurse  
}
```

## Errors

Your runbooks must handle errors. Azure Automation supports two types of PowerShell errors, terminating and non-terminating.

Terminating errors stop runbook execution when they occur. The runbook stops with a job status of Failed.

Non-terminating errors allow a script to continue even after they occur. An example of a non-terminating error is one that occurs when a runbook uses the Get-ChildItem cmdlet with a path that doesn't exist. PowerShell sees that the path doesn't exist, throws an error, and continues to the next folder. The error in this case doesn't set the runbook job status to Failed, and the job might even be completed. To force a runbook to stop on a non-terminating error, you can use ErrorAction Stop on the cmdlet.

## Calling processes

Runbooks that run in Azure sandboxes don't support calling processes, such as executables (.exe files) or subprocesses. The reason for this is that an Azure sandbox is a shared process run in a container that might not be able to access all the underlying APIs. For scenarios requiring third-party software or calls to subprocesses, you should execute a runbook on a Hybrid Runbook Worker.

## Device and application characteristics

Runbook jobs in Azure sandboxes can't access any device or application characteristics. The most common API used to query performance metrics on Windows is WMI, with some of the common metrics being memory and CPU usage. However, it doesn't matter what API is used, as jobs running in the cloud can't access the Microsoft implementation of Web-Based Enterprise Management (WBEM). This platform is built on the Common Information Model (CIM), providing the industry standards for defining device and application characteristics.

## Webhooks

External services, for example, Azure DevOps Services and GitHub, can start a runbook in Azure Automation. To do this type of startup, the service uses a webhook via a single HTTP request. Use of a webhook allows runbooks to be started without implementation of a full Azure Automation feature.

## Shared resources

To share resources among all runbooks in the cloud, Azure uses a concept called fair share. Using fair share, Azure temporarily unloads or stops any job that has run for more than three hours. Jobs for PowerShell runbooks and Python runbooks are stopped and not restarted, and the job status becomes Stopped.

For long-running Azure Automation tasks, it's recommended to use a Hybrid Runbook Worker. Hybrid Runbook Workers aren't limited by fair share, and don't have a limitation on how long a runbook can execute. The other job limits apply to both Azure sandboxes and Hybrid Runbook Workers. While Hybrid Runbook Workers aren't limited by the three-hour fair share limit, you should develop runbooks to run on the workers that support restarts from unexpected local infrastructure issues.

Another option is to optimize a runbook by using child runbooks. For example, your runbook might loop through the same function on several resources, for example, with a database operation on several databases. You can move this function to a child runbook and have your runbook call it using `Start-AzAutomationRunbook`. Child runbooks execute in parallel in separate processes.

Using child runbooks decreases the total amount of time for the parent runbook to complete. Your runbook can use the `Get-AzAutomationJob` cmdlet to check the job status for a child runbook if it still has more operations after the child completes.

## Azure Advisor

### Introduction to Azure Advisor

#### What is an Advisor?

Advisor is a personalized cloud consultant that helps you follow best practices to optimize your Azure deployments. It analyzes your resource configuration and usage telemetry and then recommends solutions that can help you improve the cost effectiveness, performance, Reliability (formerly called High availability), and security of your Azure resources.

With Advisor, you can:

- Get proactive, actionable, and personalized best practices recommendations.
- Improve the performance, security, and reliability of your resources, as you identify opportunities to reduce your overall Azure spend.
- Get recommendations with proposed actions inline.

You can access Advisor through the Azure portal. Sign in to the portal, locate Advisor in the navigation menu, or search for it in the All services menu.

The Advisor dashboard displays personalized recommendations for all your subscriptions. You can apply filters to display recommendations for specific subscriptions and resource types.

The recommendations are divided into five categories:

- Reliability (formerly called High Availability): To ensure and improve the continuity of your business-critical applications.
- Security: To detect threats and vulnerabilities that might lead to security breaches.
- Performance: To improve the speed of your applications.
- Cost: To optimize and reduce your overall Azure spending.
- Operational Excellence: To help you achieve process and workflow efficiency, resource manageability and deployment best practices.

The screenshot shows the Microsoft Azure Advisor blade. It features a sidebar with navigation links like Overview, Recommendations, Monitoring, and Settings. The main area is divided into five sections: High Availability (4 recommendations, 122 impacted resources), Security (31 recommendations, 218 impacted resources), Performance (3 recommendations, 14 impacted resources), Operational Excellence (1 recommendation, 1 impacted resource), and Cost (7,437 USD savings\*, 3 recommendations, 14 impacted resources). A purple bar at the top allows creating alerts for new recommendations.

You can click a category to display the list of recommendations within that category, and select a recommendation to learn more about it. You can also learn about actions that you can perform to take advantage of an opportunity or resolve an issue.

The screenshot shows the Microsoft Azure Advisor recommendations blade. It lists 6 High Availability recommendations. Each recommendation includes a description, potential benefits, impacted resources, and an update time. The recommendations are as follows:

Impact	Description	Potential Benefits	Impacted Resources	Updated At
Medium	Add more virtual machines for improved fault tolerance.	Ensure business continuity through virtual machine resilience.	1 Availability set	10/17/2017, 8:26:52 AM
Medium	Add more virtual machines for improved fault tolerance.	Ensure business continuity through virtual machine resilience.	2 Availability sets (classic)	10/17/2017, 12:37:27 PM
Medium	Enable virtual machine backup to protect your data from corruption and accidental deletion.	Improved data resilience and performance.	4 Virtual machines	10/17/2017, 12:39:46 PM
Medium	Use availability sets for improved fault tolerance.	Ensure business continuity through virtual machine resilience.	4 Virtual machines	10/17/2017, 8:26:40 AM
Medium	Use Premium Disks to improve I/O performance.	Improved data resilience and performance.	5 Virtual machines (classic)	10/17/2017, 12:37:47 PM
Medium	Use availability sets for improved fault tolerance.	Ensure business continuity through virtual machine resilience.	3 Virtual machines (classic)	10/17/2017, 12:39:27 PM

Select the recommended action for a recommendation to implement the recommendation. A simple interface will open that enables you to implement the recommendation or refer you to documentation that assists you with implementation. Once you implement a recommendation, it can take up to a day for an Advisor to recognize that. If you do not intend to take immediate action on a recommendation, you can postpone it for a specified time period or dismiss it. If you do not want to receive recommendations for a specific subscription or resource group, you can configure Advisor to only generate recommendations for specified subscriptions and resource groups.

## **How do I access Advisor?**

You can access Advisor through the Azure portal. Sign in to the portal, locate Advisor in the navigation menu, or search for it in the All services menu.

You can also view Advisor recommendations through the virtual machine resource interface. Choose a virtual machine, and then scroll to Advisor recommendations in the menu.

## **What permissions do I need to access Advisor?**

You can access Advisor recommendations as Owner, Contributor, or Reader of a subscription, Resource Group or Resource.

## **What resources does the Advisor provide recommendations for?**

Advisor provides recommendations for Application Gateway, App Services, availability sets, Azure Cache, Azure Data Factory, Azure Database for MySQL, Azure Database for PostgreSQL, Azure Database for MariaDB, Azure ExpressRoute, Azure Cosmos DB, Azure public IP addresses, Azure Synapse Analytics, SQL servers, storage accounts, Traffic Manager profiles, and virtual machines.

Azure Advisor also includes your recommendations from Microsoft Defender for Cloud which may include recommendations for additional resource types.

## **Can I postpone or dismiss a recommendation?**

To postpone or dismiss a recommendation, click the Postpone link. You can specify a postpone period or select Never to dismiss the recommendation.

## **Azure security baseline for Azure Advisor**

### **Identity Management**

## **IM-1: Standardize Azure Active Directory as the central identity and authentication system**

Guidance: Azure Advisor uses Azure Active Directory (Azure AD) as the default identity and access management service. Standardize Azure AD to govern your organization's identity and access management in:

- Microsoft Cloud resources, such as the Azure portal, Azure Storage, Azure Virtual Machine (Linux and Windows), Azure Key Vault, PaaS, and SaaS applications
- Your organization's resources, such as applications on Azure or your corporate network resources

Ensure securing Azure AD is a high priority in your organization's cloud security practice. Azure AD also provides an identity secure score to help you assess identity security posture relative to Microsoft's best practice recommendations. Use the score to gauge how closely your configuration matches best practice recommendations, and to make improvements in your security posture.

Note that Azure AD supports external identities, which allow users without a Microsoft account to sign in to their applications and resources with their external identity.

Responsibility: Customer

Microsoft Defender for Cloud monitoring: None

## **IM-3: Use Azure AD single sign-on (SSO) for application access**

Guidance: Azure Advisor uses Azure Active Directory (Azure AD) to provide identity and access management to Azure resources, cloud applications, and on-premises applications. This includes enterprise identities such as employees, as well as external identities such as partners, vendors, and suppliers.

Use single sign-on to manage and secure access to your organization's data and resources on-premises and in the cloud. Connect all your users, applications, and devices to the Azure AD for seamless, secure access and greater visibility and control.

Responsibility: Customer

Microsoft Defender for Cloud monitoring: None

## **Privileged Access**

### **PA-3: Review and reconcile user access regularly**

Guidance: Azure Advisor uses Azure Active Directory (Azure AD) accounts to manage its resources, review user accounts and access assignments regularly to ensure the accounts and their access are valid. Implement Azure AD access reviews to review group memberships, access to enterprise applications, and role assignments. Azure AD reporting can provide logs to help discover stale accounts.

Additionally, use Azure AD's Privileged Identity Management features to create access review report workflow to facilitate the review process. Privileged Identity Management can also be configured to alert when an excessive number of administrator accounts are created, and to identify administrator accounts that are stale or improperly configured.

Note that some Azure services support local users and roles which are not managed through Azure AD. Customers will need to manage these users separately.

Responsibility: Customer

Microsoft Defender for Cloud monitoring: None

#### **PA-6: Use privileged access workstations**

Guidance: Secured, isolated workstations are critically important for the security of sensitive roles like administrators, developers, and critical service operators.

Use highly secured user workstations and/or Azure Bastion for administrative tasks. Choose Azure Active Directory (Azure AD), Microsoft Defender Advanced Threat Protection (ATP), including Microsoft Intune to deploy a secure and managed user workstation for administrative tasks.

Centrally manage the secured workstations to enforce secured configuration including strong authentication, software and hardware baselines, restricted logical and network access.

Responsibility: Customer

Microsoft Defender for Cloud monitoring: None

#### **PA-7: Follow just enough administration (least privilege principle)**

Guidance: Azure Advisor is integrated with Azure role-based access control (Azure RBAC) to manage its resources. Use Azure RBAC to manage Azure resource access through role assignments.

Assign roles to users, groups, service principals and managed identities. There are pre-defined built-in roles for certain resources, which can be inventoried or queried through tools such as Azure CLI, Azure PowerShell or the Azure portal. The privileges assigned to resources through Azure RBAC should be always limited to what is required by the roles. This complements the just in time (JIT) approach of Azure Active Directory (Azure AD) Privileged Identity Management (PIM) and should be reviewed periodically.

Use built-in roles to allocate permission and only create custom roles when required.

What is Azure role-based access control (Azure RBAC) /azure/role-based-access-control/ overview

Responsibility: Customer

Microsoft Defender for Cloud monitoring: None

## **Asset Management**

### **AM-1: Ensure security team has visibility into risks for assets**

Guidance: Ensure security teams are granted Security Reader permissions in your Azure tenant and subscriptions so they can monitor for security risks using Microsoft Defender for Cloud.

Depending on how security team responsibilities are structured, monitoring for security risks could be the responsibility of a central security team or a local team. That said, security insights and risks must always be aggregated centrally within an organization.

Security Reader permissions can be applied broadly to an entire tenant (Root Management Group) or scoped to management groups or specific subscriptions.

Note: Additional permissions might be required to get visibility into workloads and services.

Responsibility: Customer

Microsoft Defender for Cloud monitoring: None

## **Logging and Threat Detection**

### **LT-4: Enable logging for Azure resources**

Guidance: Activity logs are automatically available and contain all write operations (PUT, POST, DELETE) for your Azure Advisor resources except read operations (GET).

Activity logs can be used to find an error when troubleshooting or to monitor how a user in your organization modified a resource.

Responsibility: Customer

Microsoft Defender for Cloud monitoring: None

### **LT-5: Centralize security log management and analysis**

Guidance: Centralised logging storage and analysis to enable correlation. For each log source, ensure you have assigned a data owner, access guidance, storage location, what tools are used to process and access the data, and data retention requirements.

Ensure you are integrating Azure activity logs into your central logging. Ingest logs via Azure Monitor to aggregate security data generated by endpoint devices, network resources, and other security systems. In Azure Monitor, use Log Analytics workspaces to query and perform analytics, and use Azure Storage accounts for long term and archival storage.

In addition, enable and onboard data to Microsoft Sentinel or a third-party SIEM. Many organizations choose to use Microsoft Sentinel for “hot” data that is used frequently and Azure Storage for “cold” data that is used less frequently.

Responsibility: Customer

Microsoft Defender for Cloud monitoring: None

## **LT-6: Configure log storage retention**

Guidance: Ensure that any storage accounts or Log Analytics workspaces used for storing Azure Advisor logs has the log retention period set according to your organization's compliance regulations. In Azure Monitor, you can set your Log Analytics workspace retention period according to your organization's compliance regulations. Use Azure Storage, Data Lake or Log Analytics workspace accounts for long-term and archival storage.

Responsibility: Customer

Microsoft Defender for Cloud monitoring: None

## **LT-7: Use approved time synchronization sources**

Guidance: Azure Advisor does not support configuring your own time synchronization sources. The Azure Advisor service relies on Microsoft time synchronization sources, and is not exposed to customers for configuration.

Responsibility: Microsoft

Microsoft Defender for Cloud monitoring: None

# Posture and Vulnerability Management

## **PV-6: Perform software vulnerability assessments**

Guidance: Azure Advisor does not expose the underlying service infrastructure to customers, and customers are unable to use their own vulnerability assessment solutions with the service. Microsoft performs vulnerability management on the underlying systems that support Azure Advisor.

Responsibility: Microsoft

Microsoft Defender for Cloud monitoring: None

## **PV-7: Rapidly and automatically remediate software vulnerabilities**

Guidance: Azure Advisor does not expose the underlying service infrastructure to customers, and customers are unable to use their own vulnerability assessment solutions with the service. Microsoft performs vulnerability management on the underlying systems that support Azure Advisor.

Responsibility: Microsoft

Microsoft Defender for Cloud monitoring: None

## PV-8: Conduct regular attack simulation

Guidance: As required, conduct penetration testing or red team activities on your Azure resources and ensure remediation of all critical security findings. Follow the Microsoft Cloud Penetration Testing Rules of Engagement to ensure your penetration tests are not in violation of Microsoft policies. Use Microsoft's strategy and execution of Red Teaming and live site penetration testing against Microsoft-managed cloud infrastructure, services, and applications.

Responsibility: Shared

Microsoft Defender for Cloud monitoring: None

## Endpoint Security

### ES-1: Use Endpoint Detection and Response (EDR)

Guidance: Azure Advisor does not expose any virtual machines or containers, which would require Endpoint Detection and Response (EDR) protection. However, the infrastructure underlying Advisor is handled by Microsoft, which includes antimalware and Endpoint Detection and Response handling.

Responsibility: Microsoft

Microsoft Defender for Cloud monitoring: None

### ES-2: Use centrally managed modern anti-malware software

Guidance: Microsoft Antimalware for Azure Cloud Services is the default antimalware for Windows Virtual Machines. For Linux Virtual Machines, use third-party anti malware solutions.

Use Microsoft Defender for Cloud's Threat detection for data services to detect malware uploaded to Azure Storage accounts.

Responsibility: Microsoft

Microsoft Defender for Cloud monitoring: None

### ES-3: Ensure anti-malware software and signatures are updated

Guidance: Microsoft Antimalware for Azure Cloud Services is the default anti-malware for Windows virtual machines (VMs). For Linux VMs, use third party antimalware solution. Also, you can use Microsoft Defender for Cloud's Threat detection for data services to detect malware uploaded to Azure Storage accounts.

The underlying infrastructure under Advisor is handled by Microsoft, which includes frequently updated antimalware software.

Responsibility: Microsoft    Microsoft Defender for Cloud monitoring: None

# Azure Arc

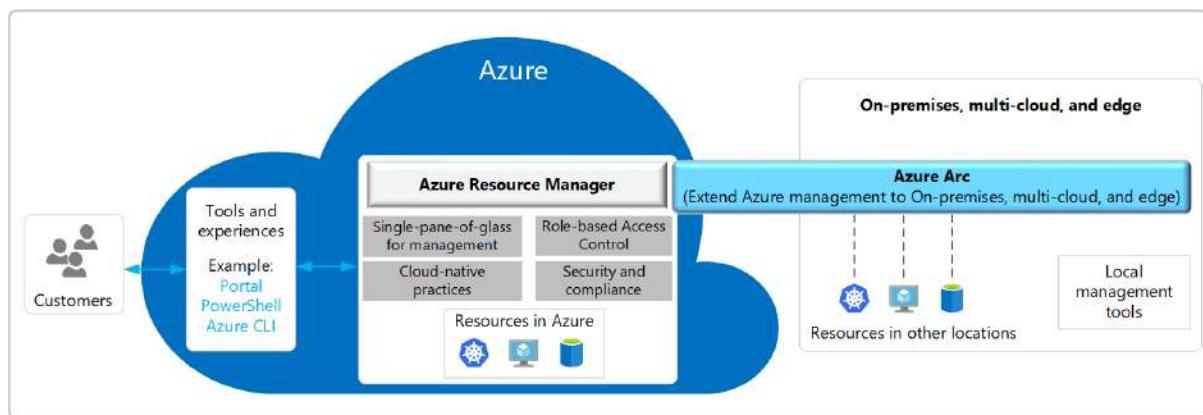
## Azure Arc overview

Today, companies struggle to control and govern increasingly complex environments. These environments extend across data centers, multiple clouds, and edge. Each environment and cloud possesses its own set of disjointed management tools that you need to learn and operate.

In parallel, new DevOps and ITOps operational models are hard to implement, as existing tools fail to provide support for new cloud native patterns.

Azure Arc simplifies governance and management by delivering a consistent multi-cloud and on-premises management platform. Azure Arc enables you to:

- Manage your entire environment, with a single pane of glass, by projecting your existing non-Azure, on-premises, or other-cloud resources into Azure Resource Manager.
- Manage virtual machines, Kubernetes clusters, and databases as if they are running in Azure.
- Use familiar Azure services and management capabilities, regardless of where they live.
- Continue using traditional ITOps, while introducing DevOps practices to support new cloud native patterns in your environment.
- Configure Custom Locations as an abstraction layer on top of Azure Arc-enabled Kubernetes cluster, cluster connect, and cluster extensions.



Today, Azure Arc allows you to manage the following resource types hosted outside of Azure:

- Servers - both physical and virtual machines running Windows or Linux.
- Kubernetes clusters - supporting multiple Kubernetes distributions.
- Azure data services - Azure SQL Managed Instance and PostgreSQL Hyperscale services.
- SQL Server - enroll instances from any location with SQL Server on Azure Arc-enabled servers.

## What does Azure Arc deliver?

Key features of Azure Arc include:

- Implement consistent inventory, management, governance, and security for your servers across your environment.
- Configure Azure VM extensions to use Azure management services to monitor, secure, and update your servers.
- Manage and govern Kubernetes clusters at scale.
- Use GitOps to deploy configuration across one or more clusters from Git repositories.
- Zero-touch compliance and configuration for your Kubernetes clusters using Azure Policy.
- Run Azure data services on any Kubernetes environment as if it runs in Azure (specifically Azure SQL Managed Instance and Azure Database for PostgreSQL Hyperscale, with benefits such as upgrades, updates, security, and monitoring). Use elastic scale and apply updates without any application downtime, even without continuous connection to Azure.
- Create custom locations on top of your Azure Arc-enabled Kubernetes clusters, using them as target locations for deploying Azure services instances. Deploy your Azure service cluster extensions for Azure Arc-enabled Data Services, App Services on Azure Arc (including web, function, and logic apps) and Event Grid on Kubernetes.
- A unified experience viewing your Azure Arc-enabled resources whether you are using the Azure portal, the Azure CLI, Azure PowerShell, or Azure REST API.

## How much does Azure Arc cost?

The following are pricing details for the features available today with Azure Arc.

### Azure Arc-enabled servers

The following Azure Arc control plane functionality is offered at no extra cost:

- Resource organization through Azure management groups and tags.
- Searching and indexing through Azure Resource Graph.
- Access and security through Azure RBAC and subscriptions.
- Environments and automation through templates and extensions.
- Update management.

Any Azure service that is used on Azure Arc-enabled servers, such as Microsoft Defender for Cloud or Azure Monitor, will be charged as per the pricing for that service.

### Azure Arc-enabled Kubernetes

Any Azure service that is used on Azure Arc-enabled Kubernetes, such as Microsoft Defender for Cloud or Azure Monitor, will be charged as per the pricing for that service.

Azure Arc-enabled data services

## What is Azure Arc-enabled servers?

Azure Arc-enabled servers enable you to manage your Windows and Linux physical servers and virtual machines hosted outside of Azure, on your corporate network, or other cloud provider. This management experience is designed to be consistent with how you manage native Azure virtual machines. When a hybrid machine is connected to Azure, it becomes a connected machine and is treated as a resource in Azure. Each connected machine has a Resource ID enabling the machine to be included in a resource group. Now you can benefit from standard Azure constructs, such as Azure Policy and applying tags. Service providers managing a customer's on-premises infrastructure can manage their hybrid machines, just like they do today with native Azure resources, across multiple customer environments using Azure Lighthouse.

To deliver this experience with your hybrid machines, you need to install the Azure Connected Machine agent on each machine. This agent does not deliver any other functionality, and it doesn't replace the Azure Log Analytics agent. The Log Analytics agent for Windows and Linux is required when:

- You want to proactively monitor the OS and workloads running on the machine,
- Manage it using Automation runbooks or solutions like Update Management, or
- Use other Azure services like Microsoft Defender for Cloud.

## Supported cloud operations

When you connect your machine to Azure Arc-enabled servers, it enables the ability for you to perform the following operational functions as described in the following table.

Operations function	Description
Govern	
Azure Policy	Assign Azure Policy guest configurations to audit settings inside the machine.
Protect	

---

**Microsoft Defender for Cloud** Protect non-Azure servers with Microsoft Defender for Endpoint, integrated through Microsoft Defender for Cloud, for threat detection, for vulnerability management, and to proactively monitor for potential security threats. Microsoft Defender for Cloud presents the alerts and remediation suggestions from the threats detected.

---

**Microsoft Sentinel** Machines connected to Arc-enabled servers can be configured with Microsoft Sentinel to collect security-related events and correlate them with other data sources.

---

## Configure

---

**Azure Automation** Automate frequent and time-consuming management tasks using PowerShell and Python runbooks.  
Assess configuration changes about installed software, Microsoft services, Windows registry and files, and Linux daemons using Change Tracking and Inventory.  
Use Update Management to manage operating system updates for Windows and Linux servers.

---

**Azure Automanage (preview)** Automate onboarding and configuration of a set of Azure services via the Automanage Machine for Arc-enabled servers.

---

**VM extensions** Provides post-deployment configuration and automation tasks using supported Arc-enabled servers VM extensions for your non-Azure Windows or Linux machine.

---

## Monitor

---

---

**Azure Monitor** Monitor the connected machine guest operating system performance and discover application components to monitor their processes and dependencies with other resources using VM insights. Collect other data, such as performance data and events, from the operating system workload(s) running on the machine with the Log Analytics agent. This data is stored in a Log Analytics workspace.

## Supported regions

In most cases, the location you select when you create the installation script should be the Azure region geographically closest to your machine's location. Data at rest is stored within the Azure geography containing the region you specify, which may also affect your choice of region if you have data residency requirements. If the Azure region your machine connects to is affected by an outage, the connected machine is not affected, but management operations using Azure may be unable to complete. If there is a regional outage, and if you have multiple locations that support a geographically redundant service, it is best to connect the machines in each location to a different Azure region.

The following metadata information about the connected machine is collected and stored in the region where the Azure Arc machine resource is configured:

- Operating system name and version
- Computer name
- Computer fully qualified domain name (FQDN)
- Connected Machine agent version

For example, if the machine is registered with Azure Arc in the East US region, this data is stored in the US region.

## Supported environments

Azure Arc-enabled servers support the management of physical servers and virtual machines hosted outside of Azure.

Note: Azure Arc-enabled servers are not designed or supported to enable management of virtual machines running in Azure.

## Agent status

The Connected Machine agent sends a regular heartbeat message to the service every 5 minutes. If the service stops receiving these heartbeat messages from a machine, that machine is considered offline and the status will automatically be changed to Disconnected in the portal within 15 to 30 minutes. Upon receiving a subsequent heartbeat message from the Connected Machine agent, its status will automatically be changed to Connected.

## Service limits

Azure Arc-enabled servers have a limit for the number of instances that can be created in each resource group. It does not have any limits at the subscription or service level.

## What is Azure Arc-enabled Kubernetes?

With Azure Arc-enabled Kubernetes, you can attach and configure Kubernetes clusters running anywhere. You can connect your clusters running on other public cloud providers (GCP, AWS) or clusters running on your on-premise data center (on VMware vSphere, Azure Stack HCI) to Azure Arc. When you connect a Kubernetes cluster to Azure Arc, it will:

- Get an Azure Resource Manager representation with a unique ID.
- Be placed in an Azure subscription and resource group.
- Receive tags just like any other Azure resource.

Azure Arc-enabled Kubernetes supports industry-standard SSL to secure data in transit. For the connected clusters, data at rest is stored encrypted in an Azure Cosmos DB database to ensure data confidentiality.

Azure Arc-enabled Kubernetes supports the following scenarios for the connected clusters:

- Connect Kubernetes running outside of Azure for inventory, grouping, and tagging.
- Deploy applications and apply configuration using GitOps-based configuration management.
- View and monitor your clusters using Azure Monitor for containers.
- Enforce threat protection using Microsoft Defender for Kubernetes.
- Apply policy definitions using Azure Policy for Kubernetes.
- Use Azure Active Directory for authentication and authorization checks on your cluster.
- Securely access your Kubernetes cluster from anywhere without opening an inbound port on the firewall using Cluster Connect.
- Deploy Open Service Mesh on top of your cluster for observability and policy enforcement on service-to-service interactions
- Deploy machine learning workloads using Azure Machine Learning for Kubernetes clusters.

- Create custom locations as target locations for deploying Azure Arc-enabled Data Services (SQL Managed Instances, PostgreSQL Hyperscale.), App Services on Azure Arc (including web, function, and logic apps) and Event Grid on Kubernetes.

Note: This service supports Azure Lighthouse, which lets service providers sign in to their own tenant to manage subscriptions and resource groups that customers have delegated.

## Supported Kubernetes distributions

Azure Arc-enabled Kubernetes works with any Cloud Native Computing Foundation (CNCF) certified Kubernetes clusters. The Azure Arc team has worked with key industry partners to validate conformance of their Kubernetes distributions with Azure Arc-enabled Kubernetes.

## What are Azure Arc-enabled data services?

Azure Arc makes it possible to run Azure data services on-premises, at the edge, and in public clouds using Kubernetes and the infrastructure of your choice.

Currently, the following Azure Arc-enabled data services are available:

- SQL Managed Instance
- PostgreSQL Hyperscale (preview)

## Always current

Azure Arc-enabled data services such as Azure Arc-enabled SQL managed instance and Azure Arc-enabled PostgreSQL Hyperscale receive updates on a frequent basis including servicing patches and new features similar to the experience in Azure. Updates from the Microsoft Container Registry are provided to you and deployment cadences are set by you in accordance with your policies. This way, on-premises databases can stay up to date while ensuring you maintain control. Because Azure Arc-enabled data services are a subscription service, you will no longer face end-of-support situations for your databases.

## Elastic scale

Cloud-like elasticity on-premises enables you to scale your databases up or down dynamically in much the same way as they do in Azure, based on the available capacity of your infrastructure. This capability can satisfy burst scenarios that have volatile needs, including scenarios that require ingesting and querying data in real time, at any scale, with sub-second response time. In addition, you can also scale out database instances using the unique hyper scale deployment option of Azure Database for PostgreSQL Hyperscale. This capability gives data workloads an additional boost on capacity optimization, using unique scale-out reads and writes.

## **Self-service provisioning**

Azure Arc also provides other cloud benefits such as fast deployment and automation at scale. Thanks to Kubernetes-based orchestration, you can deploy a database in seconds using either GUI or CLI tools.

## **Unified management**

Using familiar tools such as the Azure portal, Azure Data Studio, and the Azure CLI (az) with the arcdata extension, you can now gain a unified view of all your data assets deployed with Azure Arc. You are able to not only view and manage a variety of relational databases across your environment and Azure, but also get logs and telemetry from Kubernetes APIs to analyze the underlying infrastructure capacity and health. Besides having localized log analytics and performance monitoring, you can now leverage Azure Monitor for comprehensive operational insights across your entire estate.

## **Disconnected scenario support**

Many of the services such as self-service provisioning, automated backups/restore, and monitoring can run locally in your infrastructure with or without a direct connection to Azure. Connecting directly to Azure opens up additional options for integration with other Azure services such as Azure Monitor and the ability to use the Azure portal and Azure Resource Manager APIs from anywhere in the world to manage your Azure Arc-enabled data services.

## **Supported regions**

The following table describes the scenarios that are currently supported for Azure Arc-enabled data services.

Azure Regions	Direct connected mode	Indirect connected mode
East US	Available	Available
East US 2	Available	Available
West US	Available	Available

West US 2	Available	Available
West US 3	Available	Available
North Central US	Available	Available
Central US	Available	Available
South Central US	Available	Available
UK South	Available	Available
France Central	Available	Available
West Europe	Available	Available
North Europe	Available	Available
Japan East	Available	Available
Korea Central	Available	Available

East Asia	Available	Available
Southeast Asia	Available	Available
Australia East	Available	Available

## SQL Server on Azure Arc-enabled servers

You can manage your instances of SQL Server from Azure with SQL Server on Azure Arc-enabled servers.

You can enable SQL Server on Azure Arc-enabled servers. It extends Azure services to SQL Server instances hosted outside of Azure; in your datacenter, on the edge, or in a multi-cloud environment.

To enable Azure services, register a running SQL Server instance with Azure Arc using the Azure portal and a registration script. The registration will install a SQL Arc extension to the Connected Machine agent, which in turn will create a SQL Server – Azure Arc resource representing each SQL Server instance installed on that machine. The properties of this resource reflect a subset of the SQL Server configuration settings.

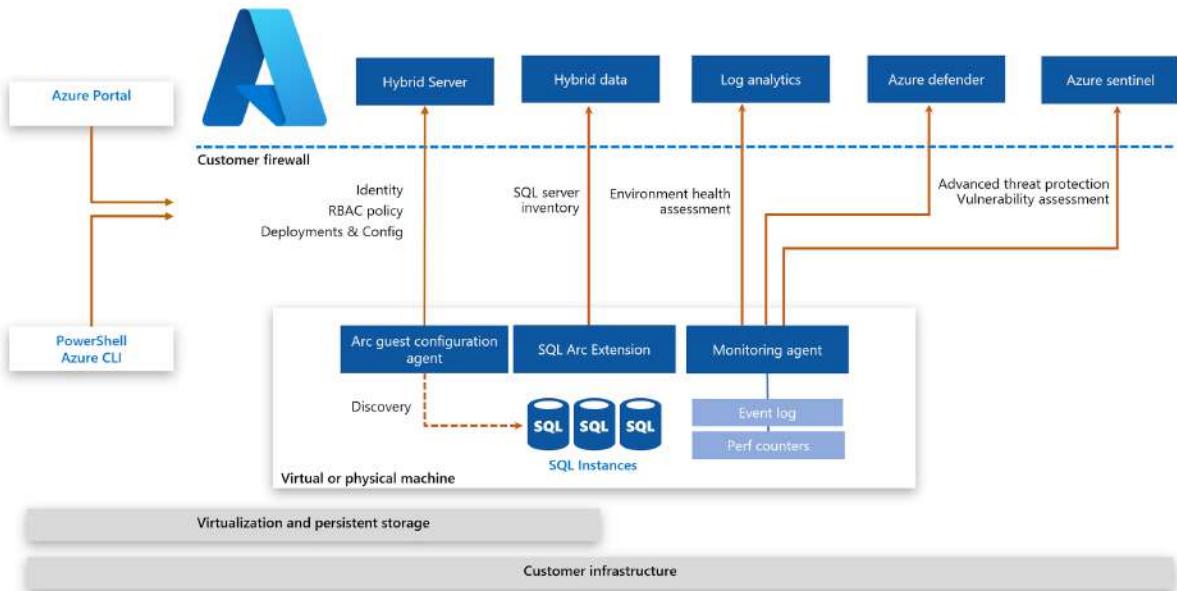
## Architecture

The SQL Server instance can be installed in a virtual or physical machine running Windows or Linux that is connected to Azure Arc via the Connected Machine agent. When you register the SQL Server instance, the agent is installed and the machine is registered automatically.

The Connected Machine agent communicates outbound securely to Azure Arc over TCP port 443. If the machine connects through a firewall or an HTTP proxy server to communicate over the Internet, review the network configuration requirements for the Connected Machine agent.

SQL Server on Azure Arc-enabled servers supports a set of solutions that require the Microsoft Monitoring Agent (MMA) server extension to be installed and connected to an Azure Log analytics workspace for data collection and reporting. These solutions include Advanced data security using Azure Security Center and Azure Sentinel, and SQL Environment health checks using On-demand SQL Assessment feature.

The following diagram illustrates the architecture of SQL Server on Azure Arc enabled servers.



## Prerequisites

### Supported SQL versions and operating systems

SQL Server on Azure Arc-enabled servers supports SQL Server 2012 or higher running on one of the following versions of the Windows or Linux operating system:

- Windows Server 2012 R2 and higher
- Ubuntu 16.04 and 18.04 (x64)
- Red Hat Enterprise Linux (RHEL) 7 (x64)
- SUSE Linux Enterprise Server (SLES) 15 (x64)

#### Note

SQL Server on Azure Arc-enabled servers does not support container images with SQL Server.

## Required permissions

To connect the SQL Server instances and the hosting machine to Azure Arc, you must have an account with privileges to perform the following actions:

- Microsoft.HybridCompute/machines/extensions/read
- Microsoft.HybridCompute/machines/extensions/write
- Microsoft.HybridCompute/machines/extensions/delete
- Microsoft.HybridCompute/machines/read
- Microsoft.HybridCompute/machines/write
- Microsoft.GuestConfiguration/guestConfigurationAssignments/read
- Microsoft.Authorization/roleAssignments/write
- Microsoft.Authorization/roleAssignments/read

For optimal security, create a custom role in Azure that has the minimal permissions listed.

## Azure subscription and service limits

Before configuring your SQL server instances and machines with Azure Arc, review the Azure Resource Manager subscription limits and resource group limits to plan for the number of machines to be connected.

## Networking configuration and resource providers

Review networking configuration, transport layer security, and resource providers required for Connected Machine agent.

The resource provider Microsoft.AzureArcData is required to connect the SQL Server instances to Azure Arc. To register the resource provider, follow the instructions in the Prerequisites section.

If you connected an instance of SQL Server to Azure Arc prior to December 2020, you need to follow the prerequisite steps to migrate the existing SQL Server - Azure Arc resources to the new namespace.

## Supported Azure regions

Arc-enabled SQL Server is available in the following regions:

- East US
- East US 2
- West US 2
- Central US
- South Central US
- UK South
- France Central
- West Europe
- North Europe
- Japan East
- Korea Central
- East Asia
- Southeast Asia
- Australia East

# Azure Automanage

## Azure Automanage for machine best practices

The following are the benefits of about Azure Automanage for machine best practices:

- Intelligently onboards virtual machines to select best practices Azure services
- Automatically configures each service per Azure best practices
- Supports customization of best practice services
- Monitors for drift and corrects for it when detected
- Provides a simple experience (point, click, set, forget)

## Overview

Azure Automanage machine best practices is a service that eliminates the need to discover, know how to onboard, and how to configure certain services in Azure that would benefit your virtual machine. These services are considered to be Azure best practices services, and help enhance reliability, security, and management for virtual machines. Example services include Azure Update Management and Azure Backup.

After onboarding your machines to Azure Automanage, each best practice service is configured to its recommended settings. However, if you want to customize the best practice services and settings, you can use the Custom Profile option.

Azure Automanage also automatically monitors for drift and corrects for it when detected. What this means is if your virtual machine or Arc-enabled server is onboarded to Azure Automanage, we'll monitor your machine to ensure that it continues to comply with its configuration profile across its entire lifecycle. If your virtual machine does drift or deviate from the profile (for example, if a service is off-boarded), we will correct it and pull your machine back into the desired state.

Automanage doesn't store/process customer data outside the geography your VMs are located. In the Southeast Asia region, Automanage does not store/process data outside of Southeast Asia.

Note: Automanage can be enabled on Azure virtual machines as well as Azure Arc-enabled servers. Automanage is not available in the US Government Cloud at this time.

## Prerequisites

There are several prerequisites to consider before trying to enable Azure AutoManage on your virtual machines.

- Supported Windows Server versions and Linux distros
- VMs must be in a supported region (see below)
- User must have correct permissions (see below)

- Automanage does not support Sandbox subscriptions at this time
- Automanage does not support Windows 10 at this time

## Supported regions

Automanage only supports VMs located in the following regions:

- West Europe
- North Europe
- Central US
- East US
- East US 2
- West US
- West US 2
- Canada Central
- West Central US
- South Central US
- Japan East
- UK South
- AU East
- AU Southeast
- Southeast Asia

## Required RBAC permissions

To onboard, Automanage requires slightly different RBAC roles depending on whether you are enabling Automanage for the first time in a subscription.

If you are enabling Automanage for the first time in a subscription:

- Owner role on the subscription(s) containing your machines, or
- Contributor and User Access Administrator roles on the subscription(s) containing your machines

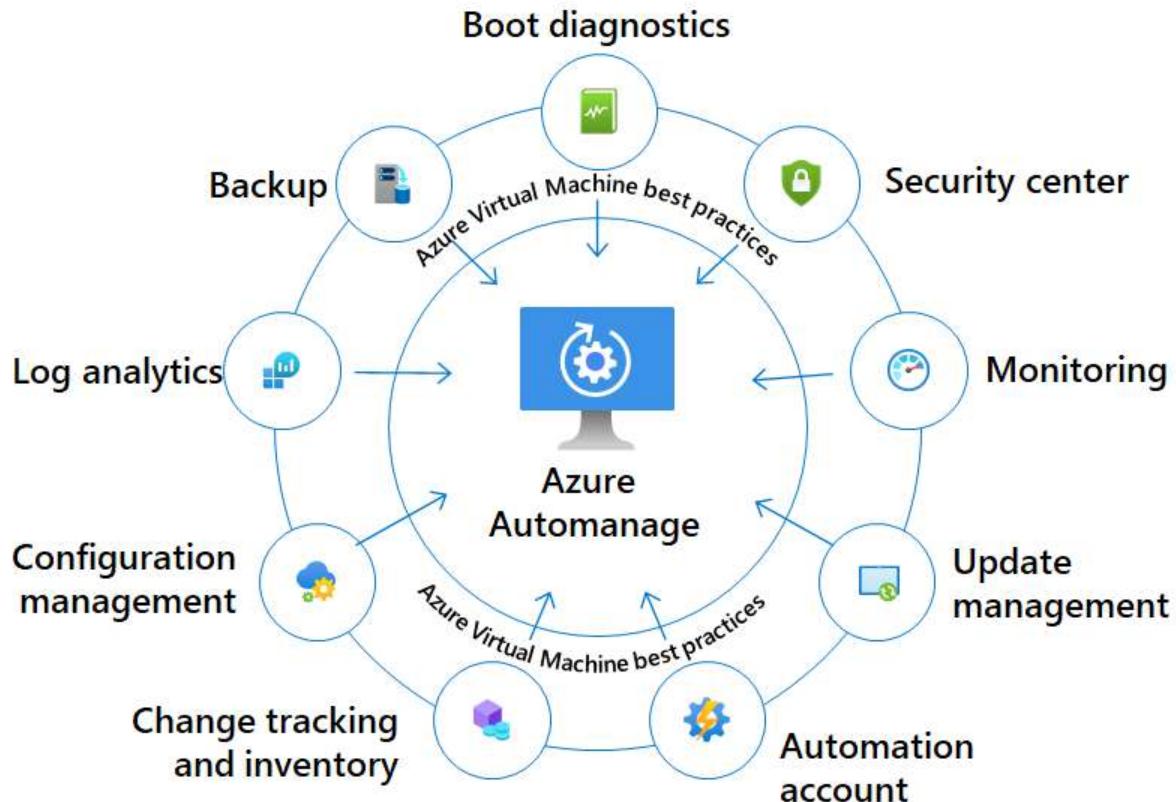
If you are enabling Automanage on a machine in a subscription that already has Automanage machines:

- Contributor role on the resource group containing your machines

The Automanage service will grant Contributor permission to this first party application (Automanage API Application Id: d828acde-4b48-47f5-a6e8-52460104a052) to perform actions on Automanaged machines. Guest users will need to have the directory reader role assigned to enable Automanage.

Note: If you want to use Automanage on a VM that is connected to a workspace in a different subscription, you must have the permissions described above on each subscription.

## Participating services



## Enabling Automanage for VMs in Azure portal

In the Azure portal, you can enable Automanage on an existing virtual machine. If it is your first time enabling Automanage for your VM, you can search in the Azure portal for Automanage – Azure machine best practices. Click Enable on the existing VM, select the configuration profile you wish to use and then select the machines you would like to onboard. Click Enable, and you're done.

The only time you might need to interact with this machine to manage these services is in the event we attempted to remediate your VM, but failed to do so. If we successfully remediate your VM, we will bring it back into compliance without even alerting you.

## Enabling Automanage for VMs using Azure Policy

You can also enable Automanage on VMs at scale using the built-in Azure Policy. The policy has a DeployIfNotExists effect, which means that all eligible VMs located within the scope of the policy will be automatically onboarded to Automanage VM Best Practices.

## Configuration Profile

When you are enabling Automanage for your machine, a configuration profile is required. Configuration profiles are the foundation of this service. They define which services

we onboard your machines to and to some extent what the configuration of those services would be.

## Best Practice Configuration Profiles

There are two best practice configuration profiles currently available.

- Dev/Test profile is designed for Dev/Test machines.
- Production profile is for production.

The reason for this differentiator is because certain services are recommended based on the workload running. For instance, in a Production machine we will automatically onboard you to Azure Backup. However, for a Dev/Test machine, a backup service would be an unnecessary cost, since Dev/Test machines are typically lower business impact.

## Custom Profiles

Custom profiles allow you to customize the services and settings that you want to apply to your machines. This is a great option if your IT requirements differ from the best practices. For instance, if you do not want to use the Microsoft Antimalware solution because your IT organization requires you to use a different antimalware solution, then you can simply toggle off Microsoft Antimalware when creating a custom profile.

Note: In the Best Practices Dev/Test configuration profile, we will not back up the VM at all.

Note: If you want to change the configuration profile of a machine, you can simply reenable it with the desired configuration profile. However, if your machine status is "Needs Upgrade" then you will need to disable first and then re enable Automanage.

## Status of VMs

In the Azure portal, go to the Automanage – Azure machine best practices page which lists all of your automanage machines. Here you will see the overall status of each machine.

Name	Resource Type	Environment	Configuration profile	Status	Operating System	Account	Subscription	Resource group
Demo-VM-8	Azure virtual machine	Dev/Test	(Custom) preference2	Configured	Windows	Automanage-Demo-Su...	Automanage-Demo-Su...	Demo-RG-4
My-VM-2	Azure virtual machine	Production	Azure Best Practices	Configured	Windows	Automanage-Demo-Su...	Automanage-Demo-Su...	My-RG-1
My-VM-3	Azure virtual machine	Production	Azure Best Practices	Configured	Linux	Automanage-Demo-Su...	Automanage-Demo-Su...	My-RG-2

For each listed machine, the following details are displayed: Name, Configuration profile, Status, Resource type, Resource group, Subscription.

The Status column can display the following states:

- In progress - the VM was just enabled and is being configured
- Conformant - the VM is configured and no drift is detected
- Not conformant - the VM has drifted and we were unable to remediate or the machine is powered off and Automanage will attempt to onboard or remediate the VM when it is next running

- Needs upgrade - the VM is onboarded to an earlier version of Automanage and needs to be upgraded to the latest version

If you see the Status as Not conformant, you can troubleshoot by clicking on the status in the portal and using the troubleshooting links provided

## Disabling Automanage for VMs

You may decide one day to disable Automanage on certain VMs. For instance, your machine is running some super sensitive secure workload and you need to lock it down even further than Azure would have done naturally, so you need to configure the machine outside of Azure best practices.

To do that in the Azure portal, go to the Automanage – Azure machine best practices page that lists all of your auto-managed VMs. Select the checkbox next to the virtual machine you want to disable from Automanage, then click on the Disable automanagement button.

Name	Resource Type	Environment	Configuration prefer...	Status	Operating System	Account	Subscription	Resource group
Demo-VM-1	Azure virtual machine	Dev/Test	(Custom) preference2	Configured	Windows	Automanage-Demo-Sub-...	Automanage-Demo-Sub-...	Demo-RG-4
My-VM-2	Azure virtual machine	Production	Azure Best Practices	Configured	Windows	Automanage-Demo-Sub-...	Automanage-Demo-Sub-...	My-RG-1

Read carefully through the messaging in the resulting pop-up before agreeing to Disable.

Note: Disabling automanagement in a VM results in the following behavior:

- The configuration of the VM and the services it is onboarded to don't change.
- Any charges incurred by those services remain billable and continue to be incurred.
- Automanage drift monitoring immediately stops.

First and foremost, we will not off-board the virtual machine from any of the services that we onboarded it to and configured. So any charges incurred by those services will continue to remain billable. You will need to off-board if necessary. Any Automanage behavior will stop immediately. For example, we will no longer monitor the VM for drift.

## Automanage and Azure Disk Encryption

Automanage is compatible with VMs that have Azure Disk Encryption (ADE) enabled. If you are using the Production environment, you will also be onboarded to Azure Backup.

## Azure Automanage for Machines Best Practices - Linux

These Azure services are automatically onboarded for you when you use Automanage Machine Best Practices Profiles on a Linux VM.

For all of these services, we will auto-onboard, auto-configure, monitor for drift, and remediate if drift is detected.

## Supported Linux distributions and versions

Automanage supports the following Linux distributions and versions:

- CentOS 7.3+, 8
- RHEL 7.4+, 8
- Ubuntu 16.04, 18.04, 20.04
- SLES 12 (SP3-SP5 only), SLES 15

## Participating services

Note: Microsoft Antimalware is not supported on Linux machines at this time.

Service	Description	Configuration Profile Supported
Machines Insights Monitoring	Azure Monitor for machines monitors the performance and health of your virtual machines, including their running processes and dependencies on other resources.	Production
Backup	Azure Backup provides independent and isolated backups to guard against unintended destruction of data on your VMs. Charges are based on the number and size of VMs being protected.	Production
Microsoft Defender for Cloud	Microsoft Defender for Cloud is a unified infrastructure security management system that strengthens the security posture of your data centers, and provides advanced threat protection across your hybrid workloads in the cloud. Automanage will configure the subscription where your VM resides to the free-tier offering of Microsoft Defender for Cloud (Enhanced security off). If your subscription is already onboarded to Microsoft Defender for Cloud, then Automanage will reconfigure it.	Production, Dev/Test

---

Update Management	You can use Update Management in Azure Automation Production, to manage operating system updates for your machines Dev/Test. You can quickly assess the status of available updates on all agent machines and manage the process of installing required updates for servers.
Change Tracking & Inventory	Change Tracking and Inventory combines change tracking and inventory functions to allow you to track virtual machine and server infrastructure changes. The service supports change tracking across services, daemons, software, registry, and files in your environment to help you diagnose unwanted changes and raise alerts. Inventory support allows you to query in-guest resources for visibility into installed applications and other configuration items.
Guest configuration	Guest configuration is used to monitor the configuration and report on the compliance of the machine. The Automanage service will install the Azure Linux baseline using the guest configuration extension. For Linux machines, the guest configuration service will install the baseline in audit-only mode. You will be able to see where your VM is out of compliance with the baseline, but noncompliance won't be automatically remediated.
Boot Diagnostics	Boot diagnostics is a debugging feature for Azure virtual machines (VM) that allows diagnosis of VM boot failures. Boot diagnostics enables a user to observe the state of their VM as it is booting up by collecting serial log information and screenshots. This will only be enabled for machines that are using managed disks.
Azure Automation Account	Azure Automation supports management throughout the lifecycle of your infrastructure and applications.

---

---

Log Analytics Workspace	Azure Monitor stores log data in a Log Analytics workspace, which is an Azure resource and a container where data is collected, aggregated, and serves as an administrative boundary.	Production, Dev/Test
-------------------------	---	----------------------

The configuration profile selection is available when you are enabling Automanage. You can also create your own custom profile with the set of Azure services and settings that you need.

## Azure Automanage for Machines Best Practices - Windows Server

These Azure services are automatically onboarded for you when you use Automanage Machine Best Practices on a Windows Server VM. For all of these services, we will auto-onboard, auto-configure, monitor for drift, and remediate if drift is detected.

### Supported Windows Server versions

Automanage supports the following Windows Server versions:

- Windows Server 2012/R2
- Windows Server 2016
- Windows Server 2019
- Windows Server 2022
- Windows Server 2022 Azure Edition

### Participating services

Service	Description	Configuration Profile
Machines Insights Monitoring	Azure Monitor for Machines monitors the performance and health of your virtual machines, including their running processes and dependencies on other resources.	Production
Backup	Azure Backup provides independent and isolated backups to guard against unintended destruction of the data on your machines.. Charges are based on the number and size of VMs being protected.	Production

---

---

Microsoft Defender for Cloud	Microsoft Defender for Cloud is a unified infrastructure security management system that strengthens the security posture of your data centers, and provides advanced threat protection across your hybrid workload in the cloud. Automanage will configure the subscription where your VM resides to the free-tier offering of Microsoft Defender for Cloud (Enhanced security off). If your subscription is already onboarded to Microsoft Defender for Cloud, then Automanage will not reconfigure it.	Production, Dev/Test
Microsoft Antimalware	Microsoft Antimalware for Azure is a free real-time protection that helps identify and remove viruses, spyware, and other malicious software. It generates alerts when known malicious or unwanted software tries to install itself or run on your Azure systems. Note: Microsoft Antimalware requires that there be no other antimalware software installed, or it may fail to work.	Production, Dev/Test
Update Management	You can use Update Management in Azure Automation to manage operating system updates for your machines. You can quickly assess the status of available updates for all agent machines and manage the process of installing required updates for servers.	Production, Dev/Test
Change Tracking & Inventory	Change Tracking and Inventory combines change tracking and inventory functions to allow you to track virtual machine and server infrastructure changes. The service supports change tracking across services, daemons, software, registry, and files in your environment to help you diagnose unwanted changes and raise alerts. Inventory support allows you to query in-guest resources for visibility into installed applications and other configuration items.	Production, Dev/Test
Guest configuration	Guest configuration policy is used to monitor the configuration and report on the compliance of the machine. The Automanage service will install the Windows security baselines using the guest configuration extension. For Windows machines, the guest configuration service will automatically reapply the baseline settings if they are out of compliance.	Production, Dev/Test

---

---

Boot Diagnostics	Boot diagnostics is a debugging feature for Azure virtual machines (VM) that allows diagnosis of VM boot failure. Boot diagnostics enables a user to observe the state of their VM as it is booting up by collecting serial log information and screenshots. This will only be enabled on machines that are using managed disks.
Windows Admin Center	Use Windows Admin Center (preview) in the Azure portal to manage the Windows Server operating system instances in an Azure VM. This is only supported for machines using Windows Server 2016 or higher. Automanage configures Windows Admin Center over a Private IP address. If you wish to connect with Windows Admin Center over a Fully Qualified Domain Name (FQDN), please open an inbound port rule for port 6516. Automanage onboards Windows Admin Center to the Dev/Test profile by default. Use the preferences to enable or disable Windows Admin Center for the Production and Dev/Test environments.
Azure Automation Account	Azure Automation supports management throughout the lifecycle of your infrastructure and applications.
Log Analytics Workspace	Azure Monitor stores log data in a Log Analytics workspace, which is an Azure resource and a container where data is collected, aggregated, and serves as an administrative boundary.

---

The configuration profile selection is available when you are enabling Automanage. You can also create your own custom profile with the set of Azure services and settings that you need.

## Azure Automanage for Machines Best Practices - Azure Arc-enabled servers

These Azure services are automatically onboarded for you when you use Automanage Machine Best Practices on an Azure Arc-enabled server VM. They are essential to our best practices white paper, which you can find in our Cloud Adoption Framework.

For all of these services, we will auto-onboard, auto-configure, monitor for drift, and remediate if drift is detected.

## Supported operating systems

Automanage supports the following operating systems for Azure Arc-enabled servers

- Windows Server 2012/R2
- Windows Server 2016
- Windows Server 2019
- CentOS 7.3+, 8
- RHEL 7.4+, 8
- Ubuntu 16.04 and 18.04
- SLES 12 (SP3-SP5 only)

## Participating services

Service	Description	Configuration Profile1
Machines Insights Monitoring	Azure Monitor for machines monitors the performance and health of your virtual machines, including their running processes and dependencies on other resources.	Production
Update Management	You can use Update Management in Azure Automation to manage operating system updates for your machines. You can quickly assess the status of available updates on all agent machines and manage the process of installing required updates for servers.	Production, Dev/Test
Change Tracking & Inventory	Change Tracking and Inventory combines change tracking and inventory functions to allow you to track virtual machine and server infrastructure changes. The service supports change tracking across services, daemons, software, registry, and files in your environment to help you diagnose unwanted changes and raise alerts. Inventory support allows you to query in-guest resources for visibility into installed applications and other configuration items.	Production, Dev/Test

---

Azure Guest Configuration	Guest Configuration policy is used to monitor the configuration and report on the compliance of the machine. The Automanage service will install the Az Linux baseline using the Guest Configuration extension. For Linux machines, the guest configuration service will install the baseline in audit-only mode. You will be able to see where your VM is out of compliance with the baseline, but noncompliance won't be automatically remediated.	Production, Dev/Test
Azure Automation Account	Azure Automation supports management throughout the lifecycle of your infrastructure and applications.	Production, Dev/Test
Log Analytics Workspace	Azure Monitor stores log data in a Log Analytics workspace, which is an Azure resource and a container where data is collected, aggregated, and serves as an administrative boundary.	Production, Dev/Test

---

The configuration profile selection is available when you are enabling Automanage. You can also create your own custom profile with the set of Azure services and settings that you need.

## Azure Backup

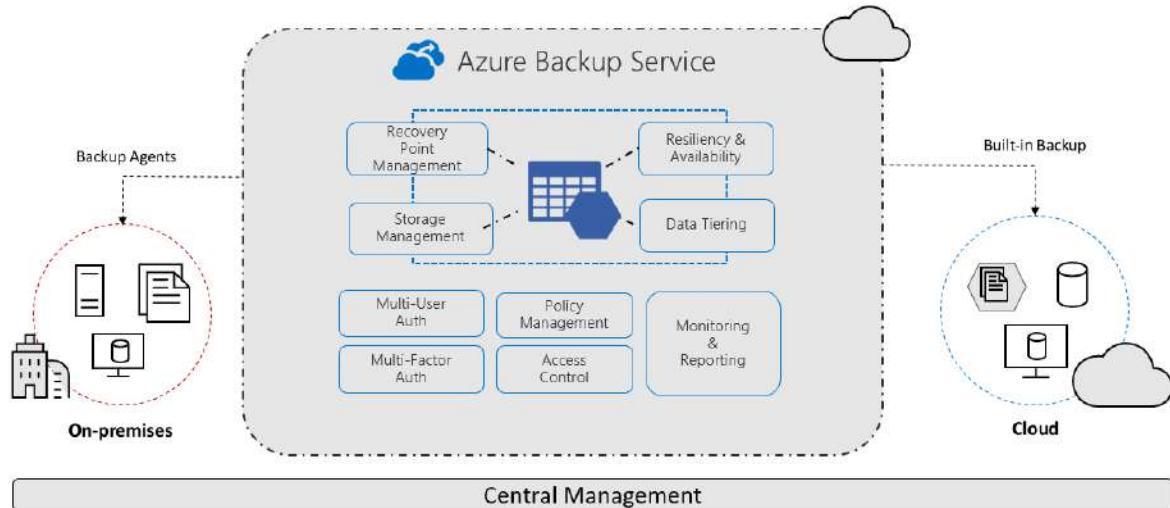
### What is the Azure Backup service?

The Azure Backup service provides simple, secure, and cost-effective solutions to back up your data and recover it from the Microsoft Azure cloud.

### What can I back up?

- On-premises - Backup files, folders, system state using the Microsoft Azure Recovery Services (MARS) agent. Or use the DPM or Azure Backup Server (MABS) agent to protect on-premises VMs (Hyper-V and VMware) and other on-premises workloads
- Azure VMs - Backup entire Windows/Linux VMs (using backup extensions) or back up files, folders, and system state using the MARS agent.
- Azure Managed Disks - Back up Azure Managed Disks
- Azure Files shares - Back up Azure File shares to a storage account
- SQL Server in Azure VMs - Back up SQL Server databases running on Azure VMs

- SAP HANA databases in Azure VMs - Backup SAP HANA databases running on Azure VMs
- Azure Database for PostgreSQL servers (preview) - Back up Azure PostgreSQL databases and retain the backups for up to 10 years
- Azure Blobs - Overview of operational backup for Azure Blobs



## Why use Azure Backup?

Azure Backup delivers these key benefits:

- Offload on-premises backup: Azure Backup offers a simple solution for backing up your on-premises resources to the cloud. Get short and long-term backup without the need to deploy complex on-premises backup solutions.
- Back up Azure IaaS VMs: Azure Backup provides independent and isolated backups to guard against accidental destruction of original data. Backups are stored in a Recovery Services vault with built-in management of recovery points. Configuration and scalability are simple, backups are optimized, and you can easily restore as needed.
- Scale easily - Azure Backup uses the underlying power and unlimited scale of the Azure cloud to deliver high-availability with no maintenance or monitoring overhead.
- Get unlimited data transfer: Azure Backup doesn't limit the amount of inbound or outbound data you transfer, or charge for the data that's transferred.
  - Outbound data refers to data transferred from a Recovery Services vault during a restore operation.
  - If you perform an offline initial backup using the Azure Import/Export service to import large amounts of data, there's a cost associated with inbound data.
- Keep data secure: Azure Backup provides solutions for securing data in transit and at rest.
- Centralized monitoring and management: Azure Backup provides built-in monitoring and alerting capabilities in a Recovery Services vault. These capabilities are available

without any additional management infrastructure. You can also increase the scale of your monitoring and reporting by using Azure Monitor.

- Get app-consistent backups: An application-consistent backup means a recovery point has all required data to restore the backup copy. Azure Backup provides application-consistent backups, which ensure additional fixes aren't required to restore the data. Restoring application-consistent data reduces the restoration time, allowing you to quickly return to a running state.
- Retain short and long-term data: You can use Recovery Services vaults for short-term and long-term data retention.
- Automatic storage management - Hybrid environments often require heterogeneous storage - some on-premises and some in the cloud. With Azure Backup, there's no cost for using on-premises storage devices. Azure Backup automatically allocates and manages backup storage, and it uses a pay-as-you-use model. So you only pay for the storage you consume.
- Multiple storage options - Azure Backup offers three types of replication to keep your storage/data highly available.
  - Locally redundant storage (LRS) replicates your data three times (it creates three copies of your data) in a storage scale unit in a datacenter. All copies of the data exist within the same region. LRS is a low-cost option for protecting your data from local hardware failures.
  - Geo-redundant storage (GRS) is the default and recommended replication option. GRS replicates your data to a secondary region (hundreds of miles away from the primary location of the source data). GRS costs more than LRS, but GRS provides a higher level of durability for your data, even if there's a regional outage.
  - Zone-redundant storage (ZRS) replicates your data in availability zones, guaranteeing data residency and resiliency in the same region. ZRS has no downtime. So your critical workloads that require data residency, and must have no downtime, can be backed up in ZRS.

## Support Matrices

### Support matrix for Azure Backup

#### Vault support

Azure Backup uses Recovery Services vaults to orchestrate and manage backups for the following workload types - Azure VMs, SQL in Azure VMs, SAP HANA in Azure VMs, Azure File shares and on-premises workloads using Azure Backup Agent, Azure Backup Server and System Center DPM. It also uses Recovery Services vaults to store backed-up data for these workloads.

The following table describes the features of Recovery Services vaults:

Feature	Details
Vaults in subscription	Up to 500 Recovery Services vaults in a single subscription.
Machines in a vault	<p>Up to 2000 data sources across all workloads (like Azure VMs, SQL Server VM, MABS Servers, and so on) can be protected in a single vault.</p> <p>Up to 1,000 Azure VMs in a single vault.</p> <p>Up to 50 MABS servers can be registered in a single vault.</p>
Data sources	Maximum size of an individual data source is 54,400 GB. This limit doesn't apply to Azure VM backups. No limits apply to the total amount of data you can back up to the vault.
Backups to vault	<p>Azure VMs: Once a day.</p> <p>Machines protected by DPM/MABS: Twice a day.</p> <p>Machines backed up directly by using the MARS agent: Three times a day.</p>
Backups between vaults	<p>Backup is within a region.</p> <p>You need a vault in every Azure region that contains VMs you want to back up. You can't back up to a different region.</p>
Move vaults	You can move vaults across subscriptions or between resource groups in the same subscription. However, moving vaults across regions is not supported.

---

Move data between vaults	Moving backed-up data between vaults isn't supported.
Modify vault storage type	You can modify the storage replication type (either geo-redundant storage or locally redundant storage) for a vault before backups are stored. After backups begin in the vault, the replication type can't be modified.
Zone-redundant storage (ZRS)	Supported in preview in UK South, South East Asia, Australia East, North Europe, Central US, East US 2, Brazil South, South Central, Korea Central, Norway East, France Central, West Europe, East, Sweden Central, Canada Central and Japan East.

---

## On-premises backup support

Here's what's supported if you want to back up on-premises machines:

Machine	What's backed up	Location	Features
Direct backup of Windows machine with MARS agent	Files, folders, system state	Backup to Recovery Services vault.	Back up three times a day No app-aware backup Restore file, folder, volume
Direct backup of Linux machine with MARS agent	Backup not supported		

---

---

Back up to DPM	Files, folders, volumes, system state, app data	Back up to local DPM storage. DPM then backs up to the vault.	App-aware snapshots  Full granularity for backup and recovery
			Linux supported for VMs (Hyper-V/ VMware)
			Oracle not supported
Back up to MABS	Files, folders, volumes, system state, app data	Back up to MABS local storage. MABS then backs up to the vault.	App-aware snapshots  Full granularity for backup and recovery
			Linux supported for VMs (Hyper-V/ VMware)
			Oracle not supported

---

## Azure VM backup support

### Azure VM limits

Limit	Details
Azure VM data disks	Support for backup of Azure VMs with up to 32 disks.  Support for backup of Azure VMs with unmanaged disks or classic disks is up to 16 disks only.
Azure VM data disk size	Individual disk size can be up to 32 TB and a maximum of 256 TB combined for all disks in a VM.

## Azure VM backup options

Here's what's supported if you want to back up Azure VMs:

Machine	What's backed up	Location	Features
Azure VM backup by using VM extension	Entire VM	Back up to the vault.	Extension installed when you enable backup for a VM.  Back up once a day.
			App-aware backup for Windows VMs; file-consistent backup for Linux VMs. You can configure app-consistency for Linux machines by using custom scripts.
			Restore VM or disk.
			Backup and restore of Active Directory domain controllers is supported.
			Can't back up an Azure VM to on-premises location.
Azure VM backup by using MARS agent	Files, folders, system state	Back up to vault.	Back up three times a day.  If you want to back up specific files or folders rather than the entire VM, the MARS agent can run alongside the VM extensio

---

Azure VM with DPM	Files, folders, volumes, system state, app data	Back up to local storage of Azure VM that's running DPM. DPM then backs up to the vault.	App-aware snapshots. Full granularity for backup and recovery.
-------------------	---	--	--

Linux supported for VMs (Hyper-V/VMware).

Oracle not supported.

---

Azure VM with MABS	Files, folders, volumes, system state, app data	Back up to local storage of Azure VM that's running MABS. MABS then backs up to the vault.	App-aware snapshots. Full granularity for backup and recovery.
--------------------	---	--	--

Linux supported for VMs (Hyper-V/VMware).

Oracle not supported.

## Linux backup support

Here's what's supported if you want to back up Linux machines:

Backup type	Linux (Azure endorsed)
-------------	------------------------

---

Direct backup of on-premises machine that's running Linux	Not supported. The MARS agent can be installed only on Windows machines.
---	--

---

Using agent extension to back up Azure VM that's running Linux	App-consistent backup by using custom script.
--	---

File-level recovery.

Restore by creating a VM from a recovery point or disk.

---

Using DPM to back up on-premises machines running Linux	File-consistent backup of Linux Guest VMs in Hyper-V and VMware.  VM restoration of Hyper-V and VMware Linux Guest VMs.
Using MABS to back up on-premises machines running Linux	File-consistent backup of Linux Guest VMs in Hyper-V and VMware.  VM restoration of Hyper-V and VMware Linux guest VMs.
<hr/>	
Using MABS or DPM to back up Linux Azure VMs	

## Daylight saving time support

Azure Backup doesn't support automatic clock adjustment for daylight saving time for Azure VM backups. It doesn't shift the hour of the backup forward or backwards. To ensure the backup runs at the desired time, modify the backup policies manually as required.

## Disk deduplication support

Disk deduplication support is as follows:

- Disk deduplication is supported on-premises when you use DPM or MABS to back up Hyper-V VMs that are running Windows. Windows Server performs data deduplication (at the host level) on virtual hard disks (VHDs) that are attached to the VM as backup storage.
- Deduplication isn't supported in Azure for any Backup component. When DPM and MABS are deployed in Azure, the storage disks attached to the VM can't be deduplicated.

## Security and encryption support

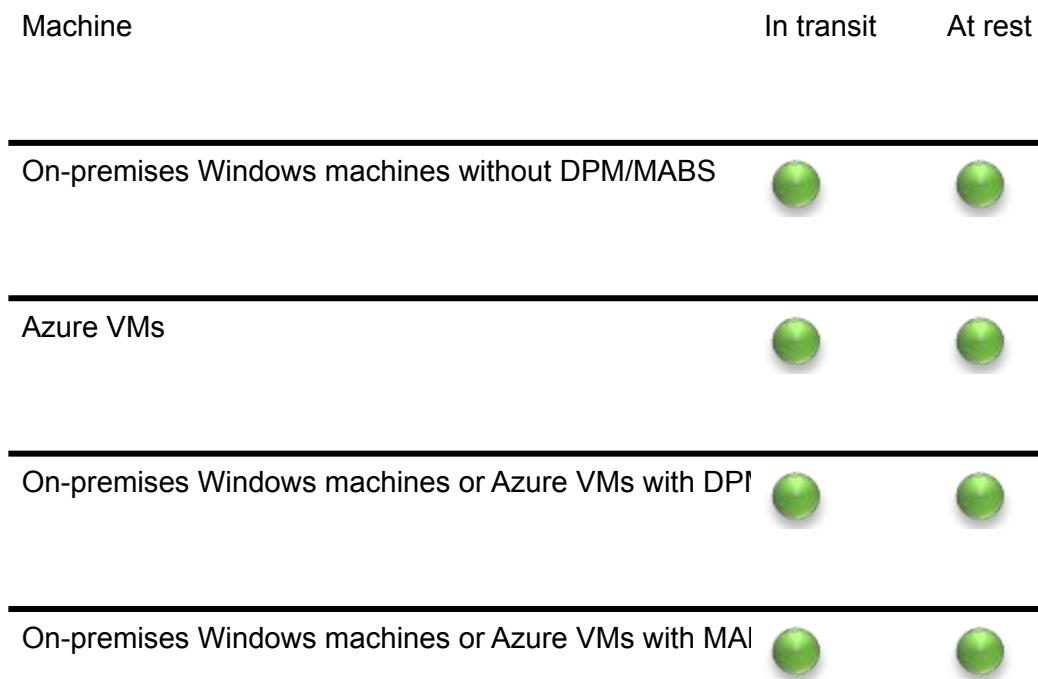
Azure Backup supports encryption for in-transit and at-rest data.

## Network traffic to Azure

- Backup traffic from servers to the Recovery Services vault is encrypted by using Advanced Encryption Standard 256.
- Backup data is sent over a secure HTTPS link.

## Data security

- Backup data is stored in the Recovery Services vault in encrypted form.
- When data is backed up from on-premises servers with the MARS agent, data is encrypted with a passphrase before upload to Azure Backup and decrypted only after it's downloaded from Azure Backup.
- When you're backing up Azure VMs, you need to set up encryption within the virtual machine.
- Azure Backup supports Azure Disk Encryption, which uses BitLocker on Windows virtual machines and dm-crypt on Linux virtual machines.
- On the back end, Azure Backup uses Azure Storage Service Encryption, which protects data at rest.



## Compression support

Backup supports the compression of backup traffic, as summarized in the following table.

- For Azure VMs, the VM extension reads the data directly from the Azure storage account over the storage network, so it isn't necessary to compress this traffic.
- If you're using DPM or MABS, you can save bandwidth by compressing the data before it's backed up.

Machine	Compress to MABS/ DPM (TCP)	Compress to vault (HTTPS)
---------	--------------------------------	------------------------------

---

Direct backup of on-premises Windows NA machines




---

Backup of Azure VMs by using VM extension

NA

---

Backup on on-premises/Azure machine by using MABS/DPM



## Retention limits

Setting	Limits
---------	--------

---

Maximum recovery points per protected instance (machine or workload)

---

Maximum expiry time for a recovery point

No limit

---

Maximum backup frequency to DPM/ MABS

Every 15 minutes for SQL Server

Once an hour for other workloads

---

Maximum backup frequency to vault	On-premises Windows machines or Azure VMs running MARS: Three per day DPM/MABS: Two per day Azure VM backup: One per day
Recovery point retention	Daily, weekly, monthly, yearly
Maximum retention period	Depends on backup frequency

---

Recovery points on DPM/MABS disk	64 for file servers; 448 for app servers Unlimited tape recovery points for on-premises DPM
----------------------------------	--

## Cross Region Restore

Azure Backup has added the Cross Region Restore feature to strengthen data availability and resiliency capability, giving you full control to restore data to a secondary region. This feature is supported for the following management types:

Backup Management type	Supported	Supported Regions
Azure VM	Supported for Azure VMs (including encrypted Azure VMs) with both managed and unmanaged disks. Not supported for classic VMs.	Available in all Azure public regions and sovereign regions except for UG IOWA and UG Virginia.

---

---

SQL /SAP HANA	Available	Available in all Azure public regions and sovereign regions except for France Central, UG IOWA, and UG Virginia.
MARS Agent/ On premises	No	N/A
AFS (Azure file shares)	No	N/A

---

## Resource health

The resource health check functions in following conditions:

Resource health check	Details
Supported Resources	Recovery Services vault
Supported Regions	East US, East US 2, Central US, South Central US, North Central US, West Central US, West US, West US 2, West US 3, Canada East, Canada Central, North Europe, West Europe, UK West, UK South, France Central, France South, Sweden Central, Sweden South, East Asia, South East Asia, Japan East, Japan West, Korea Central, Korea South, Australia East, Australia Central, Australia Central 2, Australia South East, South Africa North, South Africa West, UAE North, UAE Central, Brazil South East, Brazil South, Switzerland North, Switzerland West, Norway East, Norway West, Germany North, Germany West Central, West India, Central India, South India, Jio India West, Jio India Central.

---

---

For unsupported regions The resource health status is shown as "Unknown".

## Support matrix for Backup center

### Supported scenarios

Category	Scenario	Supported workloads	Limits
Monitoring	<ul style="list-style-type: none"><li>View all jobs</li><li>Azure Database for PostgreSQL server</li><li>SQL in Azure VM</li><li>SAP HANA in Azure VM</li><li>Azure Files</li><li>Azure Blobs</li><li>Azure Managed Disks</li></ul>	<ul style="list-style-type: none"><li>Azure Virtual Machine</li></ul>	<p>7 days worth of jobs available out of the box.</p> <p>Each filter/drop-down supports a maximum of 1000 items. So the Backup center can be used to monitor a maximum of 1000 subscriptions and 1000 vaults across tenants.</p>

---

---

Monitoring View all backup instances Azure Virtual Machine Same as above

Azure  
Database for  
PostgreSQL  
server

SQL in Azure  
VM

SAP HANA in  
Azure VM

Azure Files

Azure Blobs

Azure  
Managed  
Disks

---

---

Monitoring View all backup policies Azure Virtual Machine Same as above

Azure  
Database for  
PostgreSQL  
server

SQL in Azure  
VM

SAP HANA in  
Azure VM

Azure Files

Azure Blobs

Azure  
Managed  
Disks

---

---

Monitoring View all vaults Azure Virtual Machine Same as above

Azure  
Database for  
PostgreSQL  
server

SQL in Azure  
VM

SAP HANA in  
Azure VM

Azure Files

Azure Blobs

Azure  
Managed  
Disks

---

---

Monitoring	View Azure Monitor alerts and write metric alert rules	Azure VM	Azure Virtual Machine scale
		Azure Database for PostgreSQL server	
		SQL in Azure VM	
		SAP HANA in Azure VM	
		Azure Files	
		Azure Blobs	
		Azure Managed Disks	

---

Monitoring	View Azure Backup metrics and write metric alert rules	Azure VM	SQL in Azure VM	SAP HANA in Azure VM	Azure VM	Azure Files	You can view metrics for all Recovery Services vaults for a region and subscription simultaneously. Viewing metrics for a larger scope in the Azure portal isn't currently supported. The same limits are also applicable to configure metric alert rules.
------------	--	----------	-----------------	----------------------	----------	-------------	--

---

---

Actions	Configure backup	Azure Virtual Machine
		Azure Database for PostgreSQL server
		SQL in Azure VM
		SAP HANA in Azure VM
		Azure Files
		Azure Blobs
		Azure Managed Disks

---

---

Actions	Restore Backup	Azure Virtual Machine
		Azure Database for PostgreSQL server
		SQL in Azure VM
		SAP HANA in Azure VM
		Azure Files
		Azure Blobs
		Azure Managed Disks

---

---

Actions	Create vault	Azure Virtual Machine
		Azure Database for PostgreSQL server
		SQL in Azure VM
		SAP HANA in Azure VM
		Azure Files
		Azure Blobs
		Azure Managed Disks

---

---

Actions	Create backup policy	Azure Virtual Machine
		Azure Database for PostgreSQL server
		SQL in Azure VM
		SAP HANA in Azure VM
		Azure Files
		Azure Blobs
		Azure Managed Disks

---

---

Actions	Execute on-demand backup for a backup instance	Azure Virtual Machine
		Azure Database for PostgreSQL server
		SQL in Azure VM
		SAP HANA in Azure VM
		Azure Files
		Azure Blobs
		Azure Managed Disks

---

---

Actions	Stop backup for Azure Virtual Machine a backup instance
	Azure Database for PostgreSQL server
	SQL in Azure VM
	SAP HANA in Azure VM
	Azure Files
	Azure Blobs
	Azure Managed Disks

---

Actions	Execute cross-region restore job from Backup center	Azure Virtual Machine
		SQL in Azure VM
		SAP HANA in Azure VM

---

---

Insights	View Backup Reports	Azure Virtual Machine
		SQL in Azure Virtual Machine
		SAP HANA in Azure Virtual Machine
		Azure Files
		System Center Data Protection Manager
		Azure Backup Agent (MARS)
		Azure Backup Server (MABS)

---

Governance	View and assign built-in and custom Azure Policies under category 'Backup'	N/A	N/A
------------	--	-----	-----

---

---

Governance	View data sources not configured for backup	Azure Virtual Machine	N/A
		Azure Database for PostgreSQL server	

## Unsupported scenarios

Category	Scenario
----------	----------

---

Actions	Configuring vault settings at scale is currently not supported from Backup center
---------	---

## Support matrix for Azure VM backup

### Supported scenarios

Here's how you can back up and restore Azure VMs with the Azure Backup service.

Scenario	Backup	Agent	Restore
----------	--------	-------	---------

---

---

Direct backup of Azure VMs	Backup the entire VM.	No additional agent is needed on the Azure VM. Azure Backup installs and uses an extension to the Azure VM agent that's running on the VM.	Restore as follows: <ul style="list-style-type: none"><li>- Create a basic VM. This is useful if the VM has no special configuration such as multiple IP addresses.</li><li>- Restore the VM disk. Restore the disk. Then attach it to an existing VM, or create a new VM from the disk by using PowerShell.</li><li>- Replace VM disk. If a VM exists and it uses managed disks (unencrypted), you can restore a disk and use it to replace an existing disk on the VM.</li><li>- Restore specific files/folders. You can restore files/folders from a VM instead of from the entire VM.</li></ul>
----------------------------	-----------------------	--	---

---

Direct backup of Azure VMs (Windows only)	Back up specific files/folders/volume.	Install the Azure Recovery Services agent.  You can run the MARS agent alongside the backup extension for the Azure VM agent to back up the VM at file/folder level.	Restore specific folders/files.
---	--	--	---------------------------------

---

---

Backup Azure VM to backup server	Back up files/folders/volumes; system state/bare metal files; app data to System Center DPM to Microsoft Azure Backup Server (MABS).	Install the DPM/MABS protection agent on the VM.	The MARS agent is installed on DPM/Backup Server (MABS). MABS.	Restore files/folders/volumes; system state/bare metal files; app data
	DPM/MABS then backs up to the backup vault.			

## Supported backup actions

Action	Support
Backup a VM that's shutdown/offline	Supported. Snapshot is crash-consistent only, not app-consistent.
Back up disks after migrating to managed disks	Supported. Backup will continue to work. No action is required.
Back up managed disks after enabling resource group lock	Not supported. Azure Backup can't delete the older restore points, and backup will start to fail when the maximum limit of restore points is reached.

---

---

Modify backup policy Supported.  
for a VM

The VM will be backed up by using the schedule and retention settings in the new policy. If retention settings are extended, existing recovery points are marked and kept. If they're reduced, existing recovery points will be pruned in the next cleanup job eventually deleted.

---

Cancel a backup job Supported during the snapshot process.

Not supported when the snapshot is being transferred to the

---

Back up the VM to a different region or subscription

Not supported.  
To successfully back up, virtual machines must be in the same subscription as the vault for backup.

---

Backups per day (via the Azure VM extension)

Four backups per day - one scheduled backup as per the Backup policy, and three on-demand backups.

However, to allow user retries in case of failed attempts, the limit for on-demand backups is set to nine attempts.

---

Backups per day (via the MARS agent)

---

Backups per day (via DPM/MABS)

---

Monthly/yearly backup	<p>Not supported when backing up with Azure VM extension. Only daily and weekly is supported.</p> <p>You can set up the policy to retain daily/weekly backups for monthly/yearly retention periods.</p>
Automatic clock adjustment	<p>Not supported.</p> <p>Azure Backup doesn't automatically adjust for daylight saving changes when backing up a VM.</p> <p>Modify the policy manually as needed.</p>
Security features for hybrid backup	Disabling security features isn't supported.
Back up the VM whose machine time is changed	<p>Not supported.</p> <p>If the machine time is changed to a future date-time after enabling backup for that VM, however even if the time change is reverted, successful backup isn't guaranteed.</p>
Multiple Backups Per Day	Supported, using Enhanced policy (in preview). To enroll your subscription for this feature, write to us at <a href="mailto:askazurebackupteam@microsoft.com">askazurebackupteam@microsoft.com</a> .

## Operating system support (Windows)

The following table summarizes the supported operating systems when backing up Windows Azure VMs.

Scenario	OS support
Backup with Azure VM agent extension	<ul style="list-style-type: none"> <li>- Windows 10 Client (64 bit only)</li> <li>- Windows Server 2022 (Datacenter/Datacenter Core Standard)</li> <li>- Windows Server 2019 (Datacenter/Datacenter Core Standard)</li> <li>- Windows Server 2016 (Datacenter/Datacenter Core Standard)</li> <li>- Windows Server 2012 R2 (Datacenter/Standard)</li> <li>- Windows Server 2012 (Datacenter/Standard)</li> <li>- Windows Server 2008 R2 (RTM and SP1 Standard)</li> <li>- Windows Server 2008 (64 bit only)</li> </ul>
Back up with MARS agent	Supported operating systems.
Back up with DPM/MABS	Supported operating systems for backup with MABS and DPM.

Azure Backup doesn't support 32-bit operating systems.

## Support for Linux backup

Here's what's supported if you want to back up Linux machines.

Action	Support
Back up Linux Azure VMs with the Linux Azure VM agent	<p>File consistent backup.</p> <p>App-consistent backup using custom scripts.</p> <p>During restore, you can create a new VM, restore a disk and use it to create a VM, or restore a disk, and use it to replace a disk on an existing VM. You can also restore individual files and folders.</p>
Back up Linux Azure VMs with MARS agent	<p>Not supported.</p> <p>The MARS agent can only be installed on Windows machines.</p>
Back up Linux Azure VMs with DPM/MABS	Not supported.
Backup Linux Azure VMs with docker mounted mount points	Currently, Azure Backup doesn't support exclusion of dock mount points as these are mounted at different paths every time.

## Operating system support (Linux)

For Azure VM Linux backups, Azure Backup supports the list of Linux distributions endorsed by Azure. Note the following:

- Azure Backup doesn't support Core OS Linux.
- Azure Backup doesn't support 32-bit operating systems.
- Other bring-your-own Linux distributions might work as long as the Azure VM agent for Linux is available on the VM, and as long as Python is supported.
- Azure Backup doesn't support a proxy-configured Linux VM if it doesn't have Python version 2.7 installed.
- Azure Backup doesn't support backing up NFS files that are mounted from storage, or from any other NFS server, to Linux or Windows machines. It only backs up disks that are locally attached to the VM.

## Support matrix for managed pre-post scripts for Linux databases

Azure Backup provides support for customers to author their own pre-post scripts

Supported database	OS version	Database version
Oracle in Azure VMs	Oracle Linux	Oracle 12.x or greater

## Backup frequency and retention

Setting	Limits
Maximum recovery points per protected insta (machine/workload)	9999.
Maximum expiry time for a recovery point	No limit (99 years).
Maximum backup frequency to vault (Azure extension)	Once a day.
Maximum backup frequency to vault (MARS agent)	Three backups per day.
Maximum backup frequency to DPM/MABS	Every 15 minutes for SQL Server. Once an hour for other workloads.
Recovery point retention	Daily, weekly, monthly, and yearly.

---

Maximum retention period	Depends on backup frequency.
Recovery points on DPM/MABS disk	64 for file servers, and 448 for app servers.  Tape recovery points are unlimited for on-premises DPM.

## Supported restore methods

Restore option	Details
Create a new VM	Quickly creates and gets a basic VM up and running from a restore point.  You can specify a name for the VM, select the resource group and virtual network (VNet) in which it will be placed, and specify a storage account for the restored VM. The new VM must be created in the same region as the source VM.
Restore disk	Restores a VM disk, which can then be used to create a new VM.  Azure Backup provides a template to help you customize and create a VM.  The restore job generates a template that you can download and use to specify custom VM settings, and create a VM.  The disks are copied to the Resource Group you specify.  Alternatively, you can attach the disk to an existing VM, or create a VM using PowerShell.  This option is useful if you want to customize the VM, add configuration settings that weren't there at the time of backup, or add settings that must be configured using the template or PowerShell.

---

---

**Replace existing** You can restore a disk, and use it to replace a disk on the existing VM.

The current VM must exist. If it's been deleted, this option can't be used.

Azure Backup takes a snapshot of the existing VM before replacing the disk, and stores it in the staging location you specify. Existing disks connected to the VM are replaced with the selected restore point.

The snapshot is copied to the vault, and retained in accordance with the retention policy.

After the replace disk operation, the original disk is retained in the resource group. You can choose to manually delete the original disk if they aren't needed.

Replace existing is supported for unencrypted managed VMs and VMs created using custom images. It's not supported for unmanaged disks and VMs, classic VMs, and generalized VMs.

If the restore point has more or less disks than the current VM, the number of disks in the restore point will only reflect the VM configuration.

Replace existing is also supported for VMs with linked resources, user-assigned managed-identity and Key Vault.

---

Cross Region (secondary region)	<p>Cross Region restore can be used to restore Azure VMs in the secondary region, which is an Azure paired region.</p> <p>You can restore all the Azure VMs for the selected recovery point backup is done in the secondary region.</p> <p><b>Permissions</b> The restore operation on the secondary region can be performed by Backup Admins and App admins.</p>
---------------------------------	---

## Support for file-level restore

Restore	Supported
Restoring files across operating system	You can restore files on any machine that has the same (or compatible) OS as the backed VM.
Restoring files from encrypted VMs	Not supported.
Restoring files from network-restricted storage accounts	Not supported.
Restoring files on VMs using Windows Storage Spaces	Restore is not supported on the same VM. Instead, restore the files on a compatible VM.
Restore files on Linux VM using LVM/r arrays	Restore is not supported on the same VM. Restore on a compatible VM.
Restore files with special network settings	Restore is not supported on the same VM. Restore on a compatible VM.
Restore files from Shared disk, Temp drive, Deduplicated Disk, Ultra disk and disk with write Accelerator enabled	Restore not supported.

## Support for VM management

The following table summarizes support for backup during VM management tasks, such as adding or replacing VM disks.

Restore	Supported
Restore across subscription/region/zone.	Not supported.
Restore to an existing VM	Use the replace disk option.
Restore disk with storage account enabled for Azure Storage Service Encryption (SSE)	Not supported. Restore to an account that doesn't have SSE enabled.
Restore to mixed storage accounts	Not supported. Based on the storage account type, all resto disks will be either premium or standard, and not mixed.
Restore VM directly to an availability set	For managed disks, you can restore the disk and use the availability set option in the template.  Not supported for unmanaged disks. For unmanaged disks, restore the disk, and then create a VM in the availability set.
Restore backup of unmanaged VMs after upgrading to managed VM	Supported. You can restore disks, and then create a managed VM.

---

Restore VM to restore point before the VM was migrated to managed disks.

You restore to unmanaged disks (default), convert the restored disks to managed disk, create a VM with the managed disks.

---

Restore a VM that's been deleted.

Supported.

You can restore the VM from a recovery point.

---

Restore a domain controller VM

Supported.

---

Restore VM in different virtual network

Supported.

The virtual network must be in the same subscription and region.

## VM compute support

Compute

Support

---

VM size

Any Azure VM size with at least 2 CPU cores and 1-GB RAM

---

Back up VMs in availability sets	Supported.  You can't restore a VM in an available set by using the option to quickly create a VM. Instead, when you restore the VM, restore the disk and use it to deploy a VM, or restore a disk and use it to replace an existing disk.
Back up VMs that are supported and deployed with Hybrid Use Benefit (HUB)	
Back up VMs that are supported and deployed from Azure Marketplace	The VM must be running a supported operating system.
(Published by Microsoft, third party)	When recovering files on the VM, you can restore only to a compatible OS (not an earlier or later OS). We don't restore Marketplace VMs backed as VMs, as these need purchase information. They're only restored as disks.
Backup VMs that are supported and deployed from a custom image (third-party)	The VM must be running a supported operating system.  When recovering files on the VM, you can restore only to a compatible OS (not an earlier or later OS).
Backup VMs that are supported and migrated to Azure	To back up the VM, the VM agent must be installed on the migrated machine.
Back up Multi-VM consistency	Azure Backup doesn't provide data and application consistency across multiple VMs.

---

Backup with Diagnostic Settings	Unsupported.  If the restore of the Azure VM with diagnostic settings is triggered using the Create New option, then the restore fails.
Restore of Zone-pinned VMs	Supported (for a VM that's backed-up after Jan 2019 and where availability zones are available).  We currently support restoring to the same zone that's pinned for VMs. However, if the zone is unavailable due to an outage, the restore will fail.
Gen2 VMs	Supported Azure Backup supports backup and restore of Gen2 VMs. When these VMs are restored from Recovery point, they're restored as Gen2 VMs.
Backup of Azure VMs with locks	Unsupported for unmanaged VMs. Supported for managed VMs.
Spot VMs	Unsupported. Azure Backup restores Spot VMs as regular Azure VMs.
Azure Dedicated Hosts	Supported  While restoring an Azure VM through the Create New option though the restore gets successful, Azure VM can't be restored to the dedicated host. To achieve this, we recommend you to restore as disks. While restoring as disks with the template, create a VM in a dedicated host, and then attach the disks.  This is not applicable in the secondary region, while performing Cross Region Restore.

---

Windows Storage Spaces configuration of standalone Azure VMs	Supported
Azure Virtual Machines Scale Sets	Support for flexible orchestration model to back up and restore Single Azure VM.
Restore with Managed Identities	Yes, supported for managed Azure VMs, and not supported for classic and unmanaged Azure VMs.
Cross Region Restore	Isn't supported with managed identities.
Currently, this is available in all Azure public and national cloud regions.	
Trusted Launch VM	Backup supported (in preview)
	To enroll your subscription for this feature, write to us at <a href="mailto:askazurebackupteam@microsoft.com">askazurebackupteam@microsoft.com</a> .
	Backup for Trusted Launch VM is supported through Enhanced policy. You can enable backup only through Recovery Service vault and VM Manage blade.
	<b>Feature details</b> <ul style="list-style-type: none"><li>● Migration of an existing Generation 2 VM (protected by Azure Backup) to Trusted Launch VM is currently not supported.</li><li>● Configurations of Backup, Alerts, and Monitoring for Trusted Launch VM are currently not supported through Backup center.</li><li>● Currently, you can restore as Create VM, or Restore only.</li><li>● vTPM state doesn't persist while you restore a VM from recovery point. Therefore, scenarios that require vTPM persistence may not work across the backup and restore operation.</li></ul>

## VM storage support

Component	Support
Azure VM data disks	Support for backup of Azure VMs with up to 32 disks.  Support for backup of Azure VMs with unmanaged disks or c VMs is up to 16 disks only.
Data disk size	Individual disk size can be up to 32 TB and a maximum of 25 combined for all disks in a VM.
Storage type	Standard HDD, Standard SSD, Premium SSD.  Backup and restore of ZRS disks is supported.
Managed disks	Supported.
Encrypted disks	Supported.  Azure VMs enabled with Azure Disk Encryption can be backed up (with or without the Azure AD app).  Encrypted VMs can't be recovered at the file/folder level. You recover the entire VM.  You can enable encryption on VMs that are already protected by Azure Backup.

---

Disks with Write Accelerator enabled	Currently, Azure VM with WA disk backup is previewed in all public regions.
--------------------------------------	---

To enroll your subscription for WA Disk, write to us at [askazurebackupteam@microsoft.com](mailto:askazurebackupteam@microsoft.com).

Snapshots don't include WA disk snapshots for unsupported subscriptions as WA disk will be excluded.

**Important**

Virtual machines with WA disks need internet connectivity for successful backup (even though those disks are excluded from backup).

---

Back up & Restore deduplicated VMs/ disks	Azure Backup doesn't support deduplication. - Azure Backup doesn't deduplicate across VMs in the Recovery Services vault  - If there are VMs in deduplication state during restore, the file can't be restored because the vault doesn't understand the file. However, you can successfully perform the full VM restore.
---	---

---

Add disk to protected VM	Supported.
--------------------------	------------

---

Resize disk on protected VM	Supported.
-----------------------------	------------

---

Shared storage	Backing up VMs using Cluster Shared Volume (CSV) or Scale File Server isn't supported. CSV writers are likely to fail during backup. On restore, disks containing CSV volumes might not come-up.
----------------	--

---

Shared disks	Not supported.
--------------	----------------

---

Ultra SSD disks	Not supported.
Temporary disks	Temporary disks aren't backed up by Azure Backup.
NVMe/ephemeral disks	Not supported.
ReFS restore	Supported. VSS supports app-consistent backups on ReFS volumes like NFS.
Dynamic disk with spanned/striped volumes	<p>Supported</p> <p>If you enable selective disk features on an Azure VM, then they won't be supported.</p>

---

## VM network support

Component	Support
Number of network interfaces (NICs)	<p>Up to the maximum number of NICs supported for a specific Azure VM size.</p> <p>NICs are created when the VM is created during the restore process.</p> <p>The number of NICs on the restored VM mirrors the number of NICs on the VM when you enabled protection. Removing N after you enable protection doesn't affect the count.</p>
External/internal load balancer	Supported.

---

---

Multiple reserved IP addresses    Supported.

---

VMs with multiple network adapters    Supported.

---

VMs with public IP addresses    Supported.  
Associate an existing public IP address with the NIC, or create a new address and associate it with the NIC after restore is done.

---

Network security group (NSG) on NIC/subnet.    Supported.

---

Static IP address    Not supported.

A new VM that's created from a restore point is assigned a dynamic IP address.

For classic VMs, you can't back up a VM with a reserved IP address and no defined endpoint.

---

Dynamic IP address    Supported.

If the NIC on the source VM uses dynamic IP addressing, by default the NIC on the restored VM will use it too.

---

Azure Traffic Manager Supported.

If the backed-up VM is in Traffic Manager, manually add the restored VM to the same Traffic Manager instance.

---

Azure DNS Supported.

---

Custom DNS Supported.

---

Outbound connectivity  
via HTTP proxy

An authenticated proxy isn't supported.

---

Virtual network service endpoints Supported.

Firewall and virtual network storage account settings should allow access from all networks.

## VM security and encryption support

Azure Backup supports encryption for in-transit and at-rest data:

Network traffic to Azure:

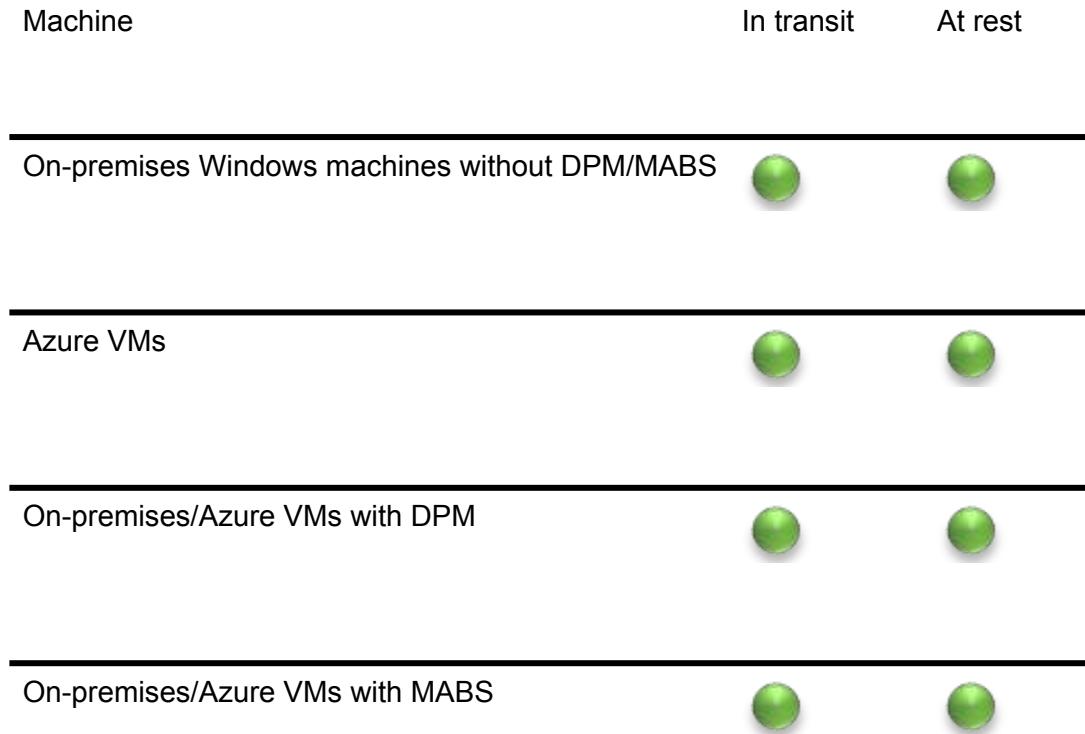
- Backup traffic from servers to the Recovery Services vault is encrypted by using Advanced Encryption Standard 256.
- Backup data is sent over a secure HTTPS link.
- The backup data is stored in the Recovery Services vault in encrypted form.
- Only you have the encryption key to unlock this data. Microsoft can't decrypt the backup data at any point.

Warning

After you set up the vault, only you have access to the encryption key. Microsoft never maintains a copy and doesn't have access to the key. If the key is misplaced, Microsoft can't recover the backup data.

## Data security:

- When backing up Azure VMs, you need to set up encryption within the virtual machine.
- Azure Backup supports Azure Disk Encryption, which uses BitLocker on Windows virtual machines and us dm-crypt on Linux virtual machines.
- On the back end, Azure Backup uses Azure Storage Service encryption, which protects data at rest.



## VM compression support

Backup supports the compression of backup traffic, as summarized in the following table. Note the following:

- For Azure VMs, the VM extension reads the data directly from the Azure storage account over the storage network. It isn't necessary to compress this traffic.
- If you're using DPM or MABS, you can save bandwidth by compressing the data before it's backed up to DPM/MABS.

Machine	Compress to MABS/DP (TCP)	Compress to vault (HTTPS)
On-premises Windows machines without DPM/MABS	NA	
Azure VMs	NA	NA
On-premises/Azure VMs with DPM		
On-premises/Azure VMs with MABS		

## Azure Backup Architecture

### What does Azure Backup do?

Azure Backup backs up the data, machine state, and workloads running on on-premises machines and Azure virtual machine (VM) instances. There are a number of Azure Backup scenarios.

### How does Azure Backup work?

You can back up machines and data by using a number of methods:

- Backup on-premises machines:
  - You can back up on-premises Windows machines directly to Azure by using the Azure Backup Microsoft Azure Recovery Services (MARS) agent. Linux machines aren't supported.
  - You can back up on-premises machines to a backup server - either System Center Data Protection Manager (DPM) or Microsoft Azure Backup Server (MABS). You can then back up the backup server to a Recovery Services vault in Azure.

- Backup Azure VMs:
  - You can back up Azure VMs directly. Azure Backup installs a backup extension to the Azure VM agent that's running on the VM. This extension backs up the entire VM.
  - You can back up specific files and folders on the Azure VM by running the MARS agent.
  - You can back up Azure VMs to the MABS that's running in Azure, and you can then back up the MABS to a Recovery Services vault.

## Where is data backed up?

Azure Backup stores backed-up data in vaults - Recovery Services vaults and Backup vaults. A vault is an online-storage entity in Azure that's used to hold data, such as backup copies, recovery points, and backup policies.

Vaults have the following features:

- Vaults make it easy to organize your backup data, while minimizing management overhead.
- You can monitor backed-up items in a vault, including Azure VMs and on-premises machines.
- You can manage vault access with Azure role-based access control (Azure RBAC).
- You specify how data in the vault is replicated for redundancy:
  - Locally redundant storage (LRS): To protect your data against server rack and drive failures, you can use LRS. LRS replicates your data three times within a single data center in the primary region. LRS provides at least 99.99999999% (11 nines) durability of objects over a given year.
  - Geo-redundant storage (GRS): To protect against region-wide outages, you can use GRS. GRS replicates your data to a secondary region.
  - Zone-redundant storage (ZRS): replicates your data in availability zones, guaranteeing data residency and resiliency in the same region.
  - By default, Recovery Services vaults use GRS.

Recovery Services vaults have the following additional features:

- In each Azure subscription, you can create up to 500 vaults.

## Backup agents

Azure Backup provides different backup agents, depending on what type of machine is being backed up:

Agent	Details
MARS agent	<ul style="list-style-type: none"> <li>Runs on individual on-premises Windows Server machine back up files, folders, and the system state.</li> <li>Runs on Azure VMs to back up files, folders, and the system state.</li> <li>Runs on DPM/MABS servers to back up the DPM/MABS I storage disk to Azure.</li> </ul>
Azure VM extension	Runs on Azure VMs to back them up to a vault.

## Backup types

The following table explains the different types of backups and when they're used:

Backup type	Details	Usage
Full	A full backup contains the entire data source. Takes more network bandwidth than differential or incremental backups.	Used for initial backup.
Differential	A differential backup stores the blocks that have changed since the initial full backup. Use a smaller amount of network and storage, as it doesn't keep redundant copies of unchanged data.  Inefficient because data blocks that are unchanged between later backups are transferred and stored.	Not used by Azure Backup.

---

Incremental	An incremental backup stores only the blocks of data that changed since the previous backup. High storage and network efficiency.	Used by DPM/MABS for disaster recovery backups, and used in all backups to Azure. Not used for SQL Server backup.
With incremental backup, there's no need to supplement with full backups.		

## SQL Server backup types

The following table explains the different types of backups used for SQL Server databases and how often they're used:

Backup type	Details	Usage
Full backup	A full database backup backs up the entire database. It contains all the data in a specific database or in a set of filegroups or files. A full backup also contains enough logs to recover that data.	At most, you can trigger one full backup per day. You can choose to make a full backup on a daily or weekly interval.
Differential backup	A differential backup is based on the most recent, previous full-data backup. It captures only the data that's changed since the full backup.	At most, you can trigger one differential backup per day. You can't configure a full backup and a differential backup on the same database.
Transaction log backup	A log backup enables point-in-time restoration up to a specific second.	At most, you can configure transactional log backups every 15 minutes.

## SAP HANA backup types

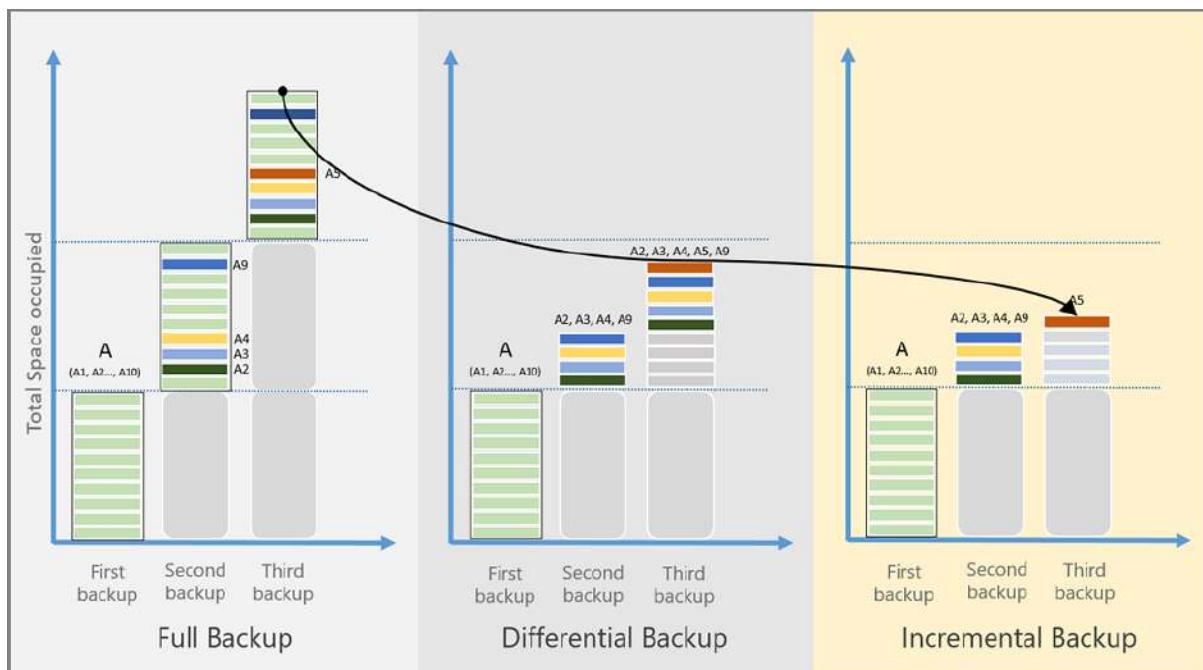
The following table explains the different types of backups used for SAP HANA databases and how often they're used:

Backup type	Details	Usage
Full backup	A full database backup backs up the entire database. This type of backup can be independently used to restore to a specific point.	You can choose to schedule a full backup on a daily or weekly interval.
Differential backup	A differential backup is based on the most recent, previous full-data backup.  It captures only the data that's changed since the previous full backup.	At most, you can schedule one differential backup per day.  You can't configure a full backup and a differential backup on the same day.
Incremental backup	An incremental backup is based on the most recent, previous full/ differential/ incremental-data backup.  It captures only the data that's changed since this previous data backup.	At most, you can schedule one incremental backup per day.  You can't schedule both differential and incremental backups on a database, only one delta backup type can be scheduled.  You can't configure a full backup and a differential backup on the same day.
Transaction log backup	A log backup enables point-in-time restoration up to a specific second.	At most, you can configure transactional log backups every minutes.

## Comparison of backup types

Storage consumption, recovery time objective (RTO), and network consumption varies for each type of backup. The following image shows a comparison of the backup types:

- Data source A is composed of 10 storage blocks, A1-A10, which are backed up monthly.
- Blocks A2, A3, A4, and A9 change in the first month, and block A5 changes in the next month.
- For differential backups, in the second month changed blocks A2, A3, A4, and A9 are backed up. In the third month, these same blocks are backed up again, along with changed block A5. The changed blocks continue to be backed up until the next full backup happens.
- For incremental backups, in the second month blocks A2, A3, A4, and A9 are marked as changed and transferred. In the third month, only changed block A5 is marked and transferred.



## Backup features

The following table summarizes the supported features for the different types of backup:

Feature	Direct Backup of Files and Folders (using MARS Agent)	Azure VM Backup	Machines or apps with DPM/MABS
---------	---	-----------------	--------------------------------

---

Back up to vault




---

---

Back up to DPM/  
MABS disk, then  
to Azure



---

Compress data  
sent for backup



No compression is used  
when transferring data.  
Storage is inflated slightly,  
but restoration is faster.



---

Run incremental  
backup



---

Back up  
deduplicated  
disks



For DPM/MABS  
servers deployed  
on-premises  
only.

Key



= Supported



= Partially Supported

<blank> = Not Supported

## Backup policy essentials

- A backup policy is created per vault.
- A backup policy can be created for the backup of following workloads: Azure VMs, SQL in Azure VMs, SAP HANA in Azure VMs and Azure file shares. The policy for files and folder backup using the MARS agent is specified in the MARS console.
  - Azure File Share
- A policy can be assigned to many resources. An Azure VM backup policy can be used to protect many Azure VMs.
- A policy consists of two components
  - Schedule: When to take the backup
  - Retention: For how long each backup should be retained.
- Schedule can be defined as "daily" or "weekly" with a specific point of time.
- Retention can be defined for "daily", "weekly", "monthly", "yearly" backup points.
  - "weekly" refers to a backup on a certain day of the week
  - "monthly" refers a backup on a certain day of the month
  - "yearly" refers to a backup on a certain day of the year

- Retention for "monthly", "yearly" backup points is referred to as Long Term Retention (LTR)
- When a vault is created, a "DefaultPolicy" is also created and can be used to back up resources.
- Any changes made to the retention period of a backup policy will be applied retroactively to all the older recovery points aside from the new ones.

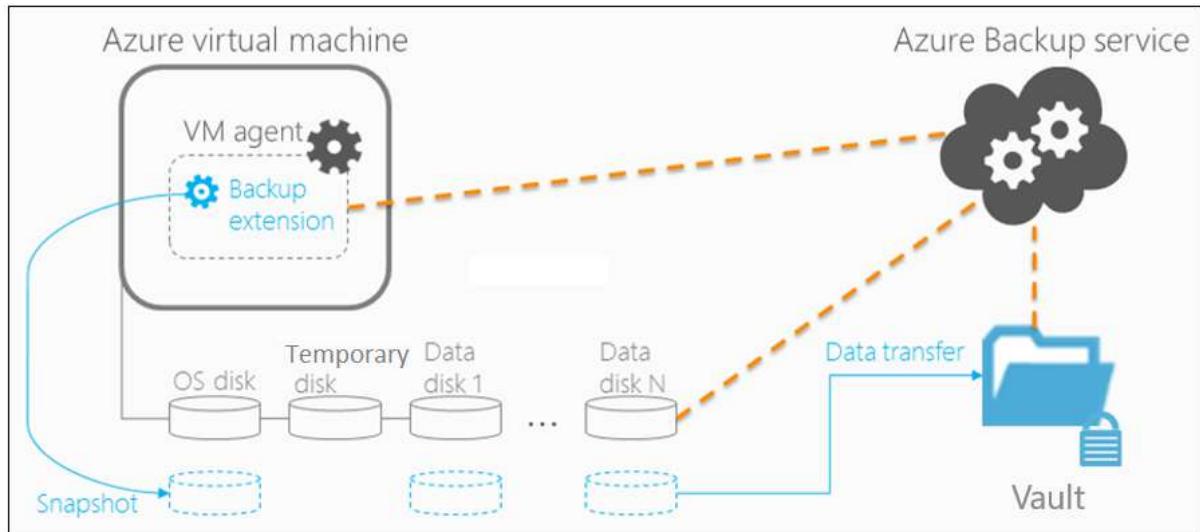
## **Impact of policy change on recovery points**

- Retention duration is increased / decreased: When the retention duration is changed, the new retention duration is applied to the existing recovery points as well. As a result, some of the recovery points will be cleaned up. If the retention period is increased, the existing recovery points will have an increased retention as well.
- Changed from daily to weekly: When the scheduled backups are changed from daily to weekly, the existing daily recovery points are cleaned up.
- Changed from weekly to daily: The existing weekly backups will be retained based on the number of days remaining according to the current retention policy.

## **Architecture: Built-in Azure VM Backup**

1. For Azure VMs that are selected for backup, Azure Backup starts a backup job according to the backup schedule you specify.
2. During the first backup, a backup extension is installed on the VM if the VM is running.
  - For Windows VMs, the VMSnapshot extension is installed.
  - For Linux VMs, the VMSnapshotLinux extension is installed.
3. For Windows VMs that are running, Backup coordinates with Windows Volume Shadow Copy Service (VSS) to take an app-consistent snapshot of the VM.
  - By default, Backup takes full VSS backups.
  - If Backup can't take an app-consistent snapshot, then it takes a file-consistent snapshot of the underlying storage (because no application writes occur while the VM is stopped).
4. For Linux VMs, Backup takes a file-consistent backup. For app-consistent snapshots, you need to manually customize pre/post scripts.
5. After Backup takes the snapshot, it transfers the data to the vault.
  - The backup is optimized by backing up each VM disk in parallel.
  - For each disk that's being backed up, Azure Backup reads the blocks on the disk and identifies and transfers only the data blocks that changed (the delta) since the previous backup.
  - Snapshot data might not be immediately copied to the vault. It might take some hours at peak times. Total backup time for a VM will be less than 24 hours for daily backup policies.

6. Changes made to a Windows VM after Azure Backup is enabled on it are:
  - Microsoft Visual C++ 2013 Redistributable(x64) - 12.0.40660 is installed in the VM
  - Startup type of Volume Shadow Copy service (VSS) changed to automatic from manual
  - IaaSVmProvider Windows service is added
7. When the data transfer is complete, the snapshot is removed, and a recovery point is created.

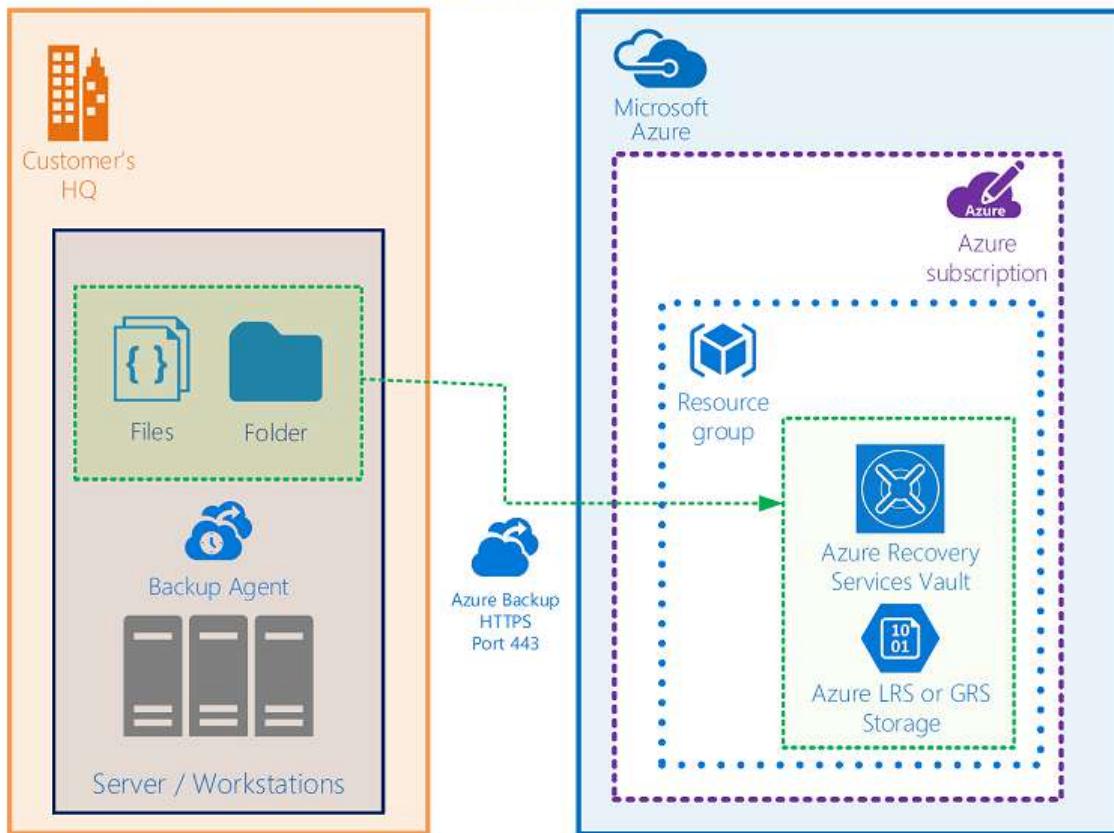


### **Architecture: Direct backup of on-premises Windows Server machines or Azure VM files or folders**

1. To set up the scenario, you download and install the MARS agent on the machine. You then select what to back up, when backups will run, and how long they'll be kept in Azure.
2. The initial backup runs according to your backup settings.
3. The MARS agent uses VSS to take a point-in-time snapshot of the volumes selected for backup.
  - The MARS agent uses only the Windows system write operation to capture the snapshot.
  - Because the agent doesn't use any application VSS writers, it doesn't capture app-consistent snapshots.
4. After taking the snapshot with VSS, the MARS agent creates a virtual hard disk (VHD) in the cache folder you specified when you configured the backup. The agent also stores checksums for each data block. These are later used to detect changed blocks for subsequent incremental backups.
5. Incremental backups run according to the schedule you specify, unless you run an on-demand backup.
6. In incremental backups, changed files are identified and a new VHD is created. The VHD is compressed and encrypted, and then it's sent to the vault.

- After the incremental backup finishes, the new VHD is merged with the VHD created after the initial replication. This merged VHD provides the latest state to be used for comparison for ongoing backup.

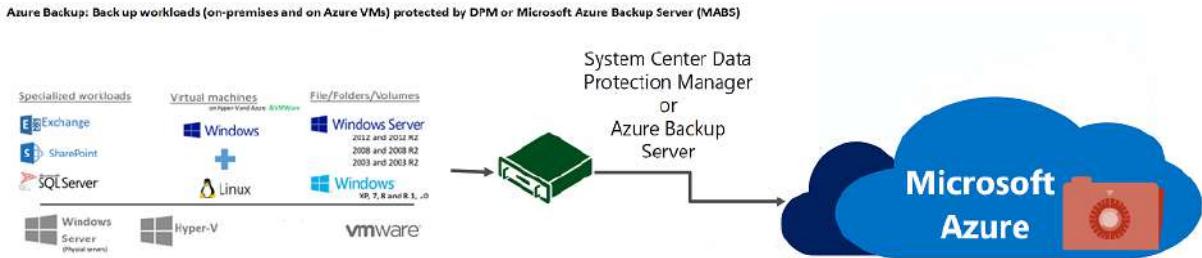
Azure Backup: Back up on-premises Windows files/folders to Azure



## Architecture: Back up to DPM/MABS

- You install the DPM or MABS protection agent on machines you want to protect. You then add the machines to a DPM protection group.
  - To protect on-premises machines, the DPM or MABS server must be located on-premises.
  - To protect Azure VMs, the MABS server must be located in Azure, running as an Azure VM.
  - With DPM/MABS, you can protect backup volumes, shares, files, and folders. You can also protect a machine's system state (bare metal), and you can protect specific apps with app-aware backup settings.
- When you set up protection for a machine or app in DPM/MABS, you select to back up to the MABS/DPM local disk for short-term storage and to Azure for online protection. You also specify when the backup to local DPM/MABS storage should run and when the online backup to Azure should run.
- The disk of the protected workload is backed up to the local MABS/DPM disks, according to the schedule you specified.

- The DPM/MABS disks are backed up to the vault by the MARS agent that's running on the DPM/MABS server.



## Azure VM storage

Azure VMs use disks to store their operating system, apps, and data. Each Azure VM has at least two disks: a disk for the operating system and a temporary disk. Azure VMs can also have data disks for app data. Disks are stored as VHDs.

- VHDs are stored as page blobs in standard or premium storage accounts in Azure:
  - Standard storage: Reliable, low-cost disk support for VMs running workloads that aren't sensitive to latency. Standard storage can use standard solid-state drive (SSD) disks or standard hard disk drive (HDD) disks.
  - Premium storage: High-performance disk support. Uses premium SSD disks.
- There are different performance tiers for disks:
  - Standard HDD disk: Backed by HDDs, and used for cost-effective storage.
  - Standard SSD disk: Combines elements of premium SSD disks and standard HDD disks. Offers more consistent performance and reliability than HDD, but still cost-effective.
  - Premium SSD disk: Backed by SSDs, and provides high-performance and low-latency for VMs that are running I/O-intensive workloads.
- Disks can be managed or unmanaged:
  - Unmanaged disks: Traditional type of disks used by VMs. For these disks, you create your own storage account and specify it when you create the disk. You then need to figure out how to maximize storage resources for your VMs.
  - Managed disks: Azure creates and manages the storage accounts for you. You specify the disk size and performance tier, and Azure creates managed disks for you. As you add disks and scale VMs, Azure handles the storage accounts.

## Backup and restore Azure VMs with premium storage

You can back up Azure VMs by using premium storage with Azure Backup:

- During the process of backing up VMs with premium storage, the Backup service creates a temporary staging location, named AzureBackup-, in the storage account. The size of the staging location equals the size of the recoverpoint snapshot.
- Make sure that the premium storage account has adequate free space to accommodate the temporary staging location. Don't modify the staging location.
- After the backup job finishes, the staging location is deleted.

- The price of storage used for the staging location is consistent with premium storage pricing.

When you restore Azure VMs by using premium storage, you can restore them to premium or standard storage. Typically, you would restore them to premium storage. But if you need only a subset of files from the VM, it might be cost effective to restore them to standard storage.

## **Backup and restore managed disks**

You can back up Azure VMs with managed disks:

- You back up VMs with managed disks in the same way that you do any other Azure VM. You can back up the VM directly from the virtual machine settings, or you can enable backup for VMs in the Recovery Services vault.
- You can back up VMs on managed disks through RestorePoint collections built on top of managed disks.
- Azure Backup also supports backing up VMs with managed disks that were encrypted by using Azure Disk Encryption.

When you restore VMs with managed disks, you can restore to a complete VM with managed disks or to a storage account:

- During the restore process, Azure handles the managed disks. If you're using the storage account option, you manage the storage account that's created during the restore process.
- If you restore a managed VM that's encrypted, make sure the VM's keys and secrets exist in the key vault before you start the restore process.

## **Azure Blueprints**

### **What is Azure Blueprints?**

Just as a blueprint allows an engineer or an architect to sketch a project's design parameters, Azure Blueprints enables cloud architects and central information technology groups to define a repeatable set of Azure resources that implements and adheres to an organization's standards, patterns, and requirements. Azure Blueprints makes it possible for development teams to rapidly build and stand up new environments with trust they're building within organizational compliance with a set of built-in components, such as networking, to speed up development and delivery.

Blueprints are a declarative way to orchestrate the deployment of various resource templates and other artifacts such as:

- Role Assignments
- Policy Assignments
- Azure Resource Manager templates (ARM templates)
- Resource Groups

The Azure Blueprints service is backed by the globally distributed Azure Cosmos DB. Blueprint objects are replicated to multiple Azure regions. This replication provides low latency, high availability, and consistent access to your blueprint objects, regardless of which region Azure Blueprints deploys your resources to.

## How it's different from ARM templates

The service is designed to help with environment setup. This setup often consists of a set of resource groups, policies, role assignments, and ARM template deployments. A blueprint is a package to bring each of these artifact types together and allow you to compose and version that package, including through a continuous integration and continuous delivery (CI/CD) pipeline. Ultimately, each is assigned to a subscription in a single operation that can be audited and tracked.

Nearly everything that you want to include for deployment in Azure Blueprints can be accomplished with an ARM template. However, an ARM template is a document that doesn't exist natively in Azure - each is stored either locally or in source control or in Templates (preview). The template gets used for deployments of one or more Azure resources, but once those resources deploy there's no active connection or relationship to the template.

With Azure Blueprints, the relationship between the blueprint definition (what should be deployed) and the blueprint assignment (what was deployed) is preserved. This connection supports improved tracking and auditing of deployments. Azure Blueprints can also upgrade several subscriptions at once that are governed by the same blueprint.

There's no need to choose between an ARM template and a blueprint. Each blueprint can consist of zero or more ARM template artifacts. This support means that previous efforts to develop and maintain a library of ARM templates are reusable in Azure Blueprints.

## How it's different from Azure Policy

A blueprint is a package or container for composing focus-specific sets of standards, patterns, and requirements related to the implementation of Azure cloud services, security, and design that can be reused to maintain consistency and compliance.

A policy is a default allow and explicit deny system focused on resource properties during deployment and for already existing resources. It supports cloud governance by validating that resources within a subscription adhere to requirements and standards.

Including a policy in a blueprint enables the creation of the right pattern or design during assignment of the blueprint. The policy inclusion makes sure that only approved or expected changes can be made to the environment to protect ongoing compliance to the intent of the blueprint.

A policy can be included as one of many artifacts in a blueprint definition. Blueprints also support using parameters with policies and initiatives.

## Blueprint definition

A blueprint is composed of artifacts. Azure Blueprints currently supports the following resources as artifacts:

Resource	Hierarchy options	Description
Resource Groups	Subscription	Create a new resource group for use by other artifacts within the blueprint. These placeholder resource groups enable you to organize resources exactly the way you want them structured and provides a scope limiter for included policy and role assignment artifacts and ARM templates.
ARM template	Subscription, Resource Group	Templates, including nested and linked templates, are used to compose complex environments. Example environments: a SharePoint farm, Azure Automation Configuration, or a Log Analytics workspace.
Policy Assignment	Subscription, Resource Group	Allows assignment of a policy or initiative to the subscription the blueprint is assigned to. The policy or initiative must be within the scope of the blueprint definition location. If the policy or initiative has parameters, these parameters are assigned at creation of the blueprint or during blueprint assignment.
Role Assignment	Subscription, Resource Group	Add an existing user or group to a built-in role to make sure the right people always have the right access to resources. Role assignments can be defined for the entire subscription or nested to a specific resource group included in the blueprint.

## Blueprint definition locations

When creating a blueprint definition, you'll define where the blueprint is saved. Blueprints can be saved to a management group or subscription that you have Contributor access to. If the location is a management group, the blueprint is available to assign to any child subscription of that management group.

## Blueprint parameters

Blueprints can pass parameters to either a policy/initiative or an ARM template. When adding either artifact to a blueprint, the author decides to provide a defined value for each blueprint assignment or to allow each blueprint assignment to provide a value at assignment time. This flexibility provides the option to define a predetermined value for all uses of the blueprint or to enable that decision to be made at the time of assignment.

Note: A blueprint can have its own parameters, but these can currently only be created if a blueprint is generated from REST API instead of through the Portal.

## Blueprint publishing

When a blueprint is first created, it's considered to be in Draft mode. When it's ready to be assigned, it needs to be Published. Publishing requires defining a Version string (letters, numbers, and hyphens with a max length of 20 characters) along with optional Change notes. The Version differentiates it from future changes to the same blueprint and allows each version to be assigned. This versioning also means different Versions of the same blueprint can be assigned to the same subscription. When additional changes are made to the blueprint, the Published Version still exists, as do the Unpublished changes. Once the changes are complete, the updated blueprint is Published with a new and unique Version and can now also be assigned.

## Blueprint assignment

Each Published Version of a blueprint can be assigned (with a max name length of 90 characters) to an existing management group or subscription. In the portal, the blueprint defaults the Version to the one Published most recently. If there are artifact parameters or blueprint parameters, then the parameters are defined during the assignment process.

Note: Assigning a blueprint definition to a management group means the assignment object exists at the management group. The deployment of artifacts still targets a subscription. To perform a management group assignment, the Create Or Update REST API must be used and the request body must include a value for properties.scope to define the target subscription.

## Permissions in Azure Blueprints

To use blueprints, you must be granted permissions through [Azure role-based access control \(Azure RBAC\)](#). To read or view a blueprint in Azure portal, your account must have read access to the scope where the blueprint definition is located.

To create blueprints, your account needs the following permissions:

- Microsoft.Blueprint/blueprints/write - Create a blueprint definition
- Microsoft.Blueprint/blueprints/artifacts/write - Create artifacts on a blueprint definition
- Microsoft.Blueprint/blueprints/versions/write - Publish a blueprint

To delete blueprints, your account needs the following permissions:

- Microsoft.Blueprint/blueprints/delete
- Microsoft.Blueprint/blueprints/artifacts/delete
- Microsoft.Blueprint/blueprints/versions/delete

Note: The blueprint definition permissions must be granted or inherited on the management group or subscription scope where it is saved.

To assign or unassign a blueprint, your account needs the following permissions:

- Microsoft.Blueprint/blueprintAssignments/write - Assign a blueprint
- Microsoft.Blueprint/blueprintAssignments/delete - Unassign a blueprint

Note: As blueprint assignments are created on a subscription, the blueprint assign and unassign permissions must be granted on a subscription scope or be inherited onto a subscription scope.

The following built-in roles are available:

Azure role	Description
Owner	In addition to other permissions, it includes all Azure Blueprints related permissions.
Contributor	In addition to other permissions, can create and delete blueprint definitions, but doesn't have blueprint assignment permissions.
Blueprint Contributor	Can manage blueprint definitions, but not assign them.
Blueprint Operator	Can assign existing published blueprints, but can't create new blueprint definitions. Blueprint assignment only works if the assignment is done with a user-assigned managed identity.

If these built-in roles don't fit your security needs, consider creating a custom role.

Note: If using a system-assigned managed identity, the service principal for Azure Blueprints requires the Owner role on the assigned subscription in order to enable deployment. If using the portal, this role is automatically granted and revoked for the deployment. If using the REST API, this role must be manually granted, but is still automatically revoked after the

deployment completes. If using a user-assigned managed identity, only the user creating the blueprint assignment needs the Microsoft.Blueprint/blueprintAssignments/write permission, which is included in both the Owner and Blueprint Operator built-in roles.

## Naming limits

The following limitations exist for certain fields:

Object	Field	Allowed Characters	Max. Length
Blueprint	Name	letters, numbers, hyphens, and periods	48
Blueprint	Version	letters, numbers, hyphens, and periods	20
Blueprint assignment	Name	letters, numbers, hyphens, and periods	90
Blueprint artifact	Name	letters, numbers, hyphens, and periods	48

## Understand the lifecycle of an Azure Blueprint

### Creating and editing a blueprint

When creating a blueprint, add artifacts to it, save to a management group or subscription, and provide a unique name and a unique version. The blueprint is now in a Draft mode and can't yet be assigned. While in the Draft mode, it can continue to be updated and changed.

A never published blueprint in Draft mode displays a different icon on the Blueprint Definitions page than ones that have been Published. The Latest Version is displayed as Draft for these never published blueprints.

Create and edit a blueprint with the Azure portal or REST API.

### Publishing a blueprint

Once all planned changes have been made to a blueprint in Draft mode, it can be Published and made available for assignment. The Published version of the blueprint can't be

altered. Once Published, the blueprint displays with a different icon than Draft blueprints and displays the provided version number in the Latest Version column.

Publish a blueprint with the Azure portal or REST API.

## Creating and editing a new version of the blueprint

A Published version of a blueprint can't be altered. However, a new version of the blueprint can be added to the existing blueprint and modified as needed. Make changes to an existing blueprint by editing it. When the new changes are saved, the blueprint now has Unpublished Changes. These changes are a new Draft version of the blueprint.

Edit a blueprint with the Azure portal.

## Publishing a new version of the blueprint

Each edited version of a blueprint must be Published before it can be assigned. When Unpublished Changes have been made to a blueprint but not Published, the Publish Blueprint button is available on the edit blueprint page. If the button isn't visible, the blueprint has already been Published and has no Unpublished Changes.

Note: A single blueprint can have multiple Published versions that can each be assigned to subscriptions. To publish a blueprint with Unpublished Changes, use the same steps for publishing a new blueprint.

## Deleting a specific version of the blueprint

Each version of a blueprint is a unique object and can be individually Published. As such, each version of a blueprint can also be deleted. Deleting a version of a blueprint doesn't have any impact on other versions of that blueprint.

Note: It's not possible to delete a blueprint that has active assignments. Delete the assignments first and then delete the version you wish to remove.

1. Select All services in the left pane. Search for and select Blueprints.
2. Select Blueprint definitions from the page on the left and use the filter options to locate the blueprint you want to delete a version of. Select it to open the edit page.
3. Select the Published versions tab and locate the version you wish to delete.
4. Right-click on the version to delete and select Delete this version.

## Deleting the blueprint

The core blueprint can also be deleted. Deleting the core blueprint also deletes any blueprint versions of that blueprint, including both Draft and Published blueprints. As with deleting a version of a blueprint, deleting the core blueprint doesn't remove the existing assignments of any of the blueprint versions.

Note: It's not possible to delete a blueprint that has active assignments. Delete the assignments first and then delete the version you wish to remove.

Delete a blueprint with the Azure portal or REST API.

## Assignments

There's several points during the lifecycle that a blueprint can be assigned to a subscription. When the mode of a version of the blueprint is Published, then that version can be assigned to a subscription. This lifecycle enables versions of a blueprint to be used and actively assigned while a newer version is being developed.

As versions of blueprints are assigned, it's important to understand where they're assigned and with what parameters they've been assigned with. The parameters can either be static or dynamic.

## Updating assignments

When a blueprint is assigned, the assignment can be updated. There are several reasons for updating an existing assignment, including:

- Add or remove resource locking
- Change the value of dynamic parameters
- Upgrade the assignment to a newer Published version of the blueprint

## Unassigning assignments

If the blueprint is no longer needed, it can be unassigned from the management group or subscription. During blueprint unassignment, the following occurs:

- Removal of blueprint resource locking
- Deletion of the blueprint assignment object
- (Conditional) If a system-assigned managed identity was used, it's also deleted

Note: All resources deployed by the blueprint assignment remain in place, but are no longer protected by Azure Blueprints.

## Stages of a blueprint deployment

### Azure Blueprints granted owner rights

The Azure Blueprints service principal is granted owner rights to the assigned subscription or subscriptions when a system-assigned managed identity is used. The granted role allows Azure Blueprints to create, and later revoke, the system-assigned managed identity. If using a user-assigned managed identity, the Azure Blueprints service principal doesn't get and doesn't need owner rights on the subscription.

The rights are granted automatically if the assignment is done through the portal. However, if the assignment is done through the REST API, granting the rights needs to be done with a

separate API call. The Azure Blueprints AppId is f71766dc-90d9-4b7d-bd9d-4499c4331c3f, but the service principal varies by tenant. Use Azure Active Directory Graph API and REST endpoint servicePrincipals to get the service principal. Then, grant the Azure Blueprints the Owner role through the Portal, Azure CLI, Azure PowerShell, REST API, or an Azure Resource Manager template.

The Azure Blueprints service doesn't directly deploy the resources.

## **The blueprint assignment object is created**

A user, group, or service principal assigns a blueprint to a subscription. The assignment object exists at the subscription level where the blueprint was assigned. Resources created by the deployment aren't done in the context of the deploying entity.

While creating the blueprint assignment, the type of managed identity is selected. The default is a system-assigned managed identity. A user-assigned managed identity can be chosen. When using a user-assigned managed identity, it must be defined and granted permissions before the blueprint assignment is created. Both the Owner and Blueprint Operator built-in roles have the necessary blueprintAssignment/write permission to create an assignment that uses a user-assigned managed identity.

## **Optional - Azure Blueprints creates system-assigned managed identity**

When system-assigned managed identity is selected during assignment, Azure Blueprints creates the identity and grants the managed identity the owner role. If an existing assignment is upgraded, Azure Blueprints uses the previously created managed identity.

The managed identity related to the blueprint assignment is used to deploy or redeploy the resources defined in the blueprint. This design avoids assignments inadvertently interfering with each other. This design also supports the resource locking feature by controlling the security of each deployed resource from the blueprint.

## **The managed identity deploys blueprint artifacts**

The managed identity then triggers the Resource Manager deployments of the artifacts within the blueprint in the defined sequencing order. The order can be adjusted to ensure artifacts dependent on other artifacts are deployed in the correct order.

An access failure by a deployment is often the result of the level of access granted to the managed identity. The Azure Blueprints service manages the security lifecycle of the system-assigned managed identity. However, the user is responsible for managing the rights and lifecycle of a user-assigned managed identity.

## **Blueprint service and system-assigned managed identity rights are revoked**

Once the deployments are completed, Azure Blueprints revokes the rights of the system-assigned managed identity from the subscription. Then, the Azure Blueprints service revokes its rights from the subscription. Rights removal prevents Azure Blueprints from becoming a permanent owner on a subscription.

## **Creating dynamic blueprints through parameters**

A fully defined blueprint with various artifacts such as resource groups, Azure Resource Manager templates (ARM templates), policies, or role assignments, offers the rapid creation and consistent creation of objects within Azure. To enable flexible use of these reusable design patterns and containers, Azure Blueprints supports parameters. The parameter creates flexibility, both during definition and assignment, to change properties on the artifacts deployed by the blueprint.

A simple example is the resource group artifact. When a resource group is created, it has two required values that must be provided: name and location. When adding a resource group to your blueprint, if parameters didn't exist, you would define that name and location for every use of the blueprint. This repetition would cause every use of the blueprint to create artifacts in the same resource group. Resources inside that resource group would become duplicated and cause a conflict.

**Note:** It isn't an issue for two different blueprints to include a resource group with the same name. If a resource group included in a blueprint already exists, the blueprint continues to create the related artifacts in that resource group. This could cause a conflict as two resources with the same name and resource type cannot exist within a subscription.

The solution to this problem is parameters. Azure Blueprints allows you to define the value for each property of the artifact during assignment to a subscription. The parameter makes it possible to reuse a blueprint that creates a resource group and other resources within a single subscription without having conflict.

## **Blueprint parameters**

Through the REST API, parameters can be created on the blueprint itself. These parameters are different than the parameters on each of the supported artifacts. When a parameter is created on the blueprint, it can be used by the artifacts in that blueprint. An example might be the prefix for naming of the resource group. The artifact can use the blueprint parameter to create a "mostly dynamic" parameter. As the parameter can also be defined during assignment, this pattern allows for a consistency that may adhere to naming rules.

## Using `secureString` and `secureObject` parameters

While an ARM template artifact supports parameters of the `secureString` and `secureObject` types, Azure Blueprints requires each to be connected with an Azure Key Vault. This security measure prevents the unsafe practice of storing secrets along with the Blueprint and encourages employment of secure patterns. Azure Blueprints supports this security measure, detecting the inclusion of either secure parameter in an ARM template artifact. The service then prompts during assignment for the following Key Vault properties per detected secure parameter:

- Key Vault resource ID
- Key Vault secret name
- Key Vault secret version

If the blueprint assignment uses a system-assigned managed identity, the referenced Key Vault must exist in the same subscription the blueprint definition is assigned to.

If the blueprint assignment uses a user-assigned managed identity, the referenced Key Vault may exist in a centralized subscription. The managed identity must be granted appropriate rights on the Key Vault prior to blueprint assignment.

### Important

In both cases, the Key Vault must have Enable access to Azure Resource Manager for template deployment configured on the Access policies page.

## Parameter types

### Static parameters

A parameter value defined in the definition of a blueprint is called a static parameter, because every use of the blueprint will deploy the artifact using that static value. In the resource group example, while it doesn't make sense for the name of the resource group, it might make sense for the location. Then, every assignment of the blueprint would create the resource group, whatever it's called during assignment, in the same location. This flexibility allows you to be selective in what you define as required vs what can be changed during assignment.

### Setting static parameters in the portal

1. Select All services in the left pane. Search for and select Blueprints.
2. Select Blueprint definitions from the page on the left.
3. Select an existing blueprint and then select Edit blueprint OR select + Create blueprint and fill out the information on the Basics tab.
4. Select Next: Artifacts OR select the Artifacts tab.
5. Artifacts added to the blueprint that have parameter options display X of Y parameters populated in the Parameters column. Select the artifact row to edit the artifact

parameters.

Role assignment	1 out of 1 parameters populated
Policy assignment	None
Resource group	1 out of 2 parameters populated

6. The Edit Artifact page displays value options appropriate to the artifact selected. Each parameter on the artifact has a title, a value box, and a checkbox. Set the box to unchecked to make it a static parameter. In the following example, only Location is a static parameter as it's unchecked and Resource Group Name is checked.

Resource Group Name  
  
 This value should be specified when the blueprint is assigned

Location  
  
 This value should be specified when the blueprint is assigned

Resource Group Tags (Optional):

TAG NAME	TAG VALUE	
<input type="text" value="Enter tag name"/>	:	<input type="text"/>

### Setting static parameters from REST API

In each REST API URI, there are variables that are used that you need to replace with your own values:

- {YourMG} - Replace with the name of your management group
- {subscriptionId} - Replace with your subscription ID

### *Blueprint level parameter*

When creating a blueprint through REST API, it's possible to create [blueprint parameters](#). To do so, use the following REST API URI and body format:

- REST API URI

```

PUT https://management.azure.com/providers/Microsoft.Management/
managementGroups/{YourMG}/providers/Microsoft.Blueprint/blueprints/
MyBlueprint?api-version=2018-11-01-preview
    ● Request Body
{
    "properties": {
        "description": "This blueprint has blueprint level parameters.",
        "targetScope": "subscription",
        "parameters": {
            "owners": {
                "type": "array",
                "metadata": {
                    "description": "List of AAD object IDs that is assigned Owner role at the
resource group"
                }
            }
        },
        "resourceGroups": {
            "storageRG": {
                "description": "Contains the resource template deployment and a role
assignment."
            }
        }
    }
}

```

Once a blueprint level parameter is created, it can be used on artifacts added to that blueprint. The following REST API example creates a role assignment artifact on the blueprint and uses the blueprint level parameter.

- REST API URI

```

PUT https://management.azure.com/providers/Microsoft.Management/
managementGroups/{YourMG}/providers/Microsoft.Blueprint/blueprints/
MyBlueprint/artifacts/roleOwner?api-version=2018-11-01-preview
    ● Request Body
{

```

```

    "kind": "roleAssignment",
    "properties": {
        "resourceGroup": "storageRG",

```

```

        "roleDefinitionId": "/providers/Microsoft.Authorization/roleDefinitions/
8e3af657-a8ff-443c-a75c-2fe8c4bcb635",
        "principalIds": "[parameters('owners')]"
    }
}

```

In this example, the principalIds property uses the owners blueprint level parameter by using a value of [parameters('owners')]. Setting a parameter on an artifact using a blueprint level parameter is still an example of a static parameter. The blueprint level parameter can't be set during blueprint assignment and will be the same value on each assignment.

### *Artifact level parameter*

Creating static parameters on an artifact is similar, but takes a straight value instead of using the parameters() function. The following example creates two static parameters, tagName and tagValue. The value on each is directly provided and doesn't use a function call.

- REST API URI

```
PUT https://management.azure.com/providers/Microsoft.Management/
managementGroups/{YourMG}/providers/Microsoft.Blueprint/blueprints/
MyBlueprint/artifacts/policyStorageTags?api-version=2018-11-01-preview
```

- Request Body

```
{
    "kind": "policyAssignment",
    "properties": {
        "description": "Apply storage tag and the parameter also used by the template to
resource groups",
        "policyDefinitionId": "/providers/Microsoft.Authorization/policyDefinitions/
49c88fc8-6fd1-46fd-a676-f12d1d3a4c71",
        "parameters": {
            "tagName": {
                "value": "StorageType"
            },
            "tagValue": {
                "value": "Premium_LRS"
            }
        }
    }
}
```

## Dynamic parameters

The opposite of a static parameter is a dynamic parameter. This parameter isn't defined on the blueprint, but instead is defined during each assignment of the blueprint. In the resource group example, use of a dynamic parameter makes sense for the resource group name. It provides a different name for every assignment of the blueprint.

### Setting dynamic parameters in the portal

1. Select All services in the left pane. Search for and select Blueprints.
2. Select Blueprint definitions from the page on the left.
3. Right-click on the blueprint that you want to assign. Select Assign blueprint OR select the blueprint you want to assign, then use the Assign blueprint button.
4. On the Assign blueprint page, find the Artifact parameters section. Each artifact with at least one dynamic parameter displays the artifact and the configuration options. Provide required values to the parameters before assigning the blueprint. In the following example, Name is a dynamic parameter that must be defined to complete blueprint assignment.

Artifact parameters	
ARTIFACT / PARAMETER	PARAMETER VALUE
Subscription	
ResourceGroup	
Resource Group: Name	<input type="button" value="Set value(s)"/>
Resource Group: Location	<input type="text" value="eastus"/>

### Setting dynamic parameters from REST API

Setting dynamic parameters during the assignment is done by entering the value directly. Instead of using a function, such as parameters(), the value provided is an appropriate string. Artifacts for a resource group are defined with a "template name", name, and location properties. All other parameters for the included artifact are defined under parameters with a <name> and value key pair. If the blueprint is configured for a dynamic parameter that isn't provided during assignment, the assignment will fail.

- REST API URI

PUT <https://management.azure.com/subscriptions/{subscriptionId}/providers/Microsoft.Blueprint/blueprintAssignments/assignMyBlueprint?api-version=2018-11-01-preview>

- Request Body

{

  "properties": {

```
        "blueprintId": "/providers/Microsoft.Management/managementGroups/{YourMG} /providers/Microsoft.Blueprint/blueprints/MyBlueprint",
        "resourceGroups": {
            "storageRG": {
                "name": "StorageAccount",
                "location": "eastus2"
            }
        },
        "parameters": {
            "storageAccountType": {
                "value": "Standard_GRS"
            },
            "tagName": {
                "value": "CostCenter"
            },
            "tagValue": {
                "value": "ContosoIT"
            },
            "contributors": {
                "value": [
                    "7be2f100-3af5-4c15-bcb7-27ee43784a1f",
                    "38833b56-194d-420b-90ce-cff578296714"
                ]
            },
            "owners": {
                "value": [
                    "44254d2b-a0c7-405f-959c-f829ee31c2e7",
                    "316deb5f-7187-4512-9dd4-21e7798b0ef9"
                ]
            }
        },
        "identity": {
            "type": "systemAssigned"
        },
        "location": "westus"
    }
}
```

# Understand the deployment sequence in Azure Blueprints

Azure Blueprints uses a sequencing order to determine the order of resource creation when processing the assignment of a blueprint definition. This article explains the following concepts:

- The default sequencing order that is used
- How to customize the order
- How the customized order is processed

There are variables in the JSON examples that you need to replace with your own values:

- {YourMG} - Replace with the name of your management group

## Default sequencing order

If the blueprint definition contains no directive for the order to deploy artifacts or the directive is null, then the following order is used:

- Subscription level role assignment artifacts sorted by artifact name
- Subscription level policy assignment artifacts sorted by artifact name
- Subscription level Azure Resource Manager template (ARM templates) artifacts sorted by artifact name
- Resource group artifacts (including child artifacts) sorted by placeholder name

Within each resource group artifact, the following sequence order is used for artifacts to be created within that resource group:

- Resource group child role assignment artifacts sorted by artifact name
- Resource group child policy assignment artifacts sorted by artifact name
- Resource group child Azure Resource Manager template (ARM templates) artifacts sorted by artifact name

### Note

Use of artifacts() creates an implicit dependency on the artifact being referred to.

## Customizing the sequencing order

When composing large blueprint definitions, it may be necessary for resources to be created in a specific order. The most common use pattern of this scenario is when a blueprint definition includes several ARM templates. Azure Blueprints handles this pattern by allowing the sequencing order to be defined.

The ordering is accomplished by defining a dependsOn property in the JSON. The blueprint definition, for resource groups, and artifact objects support this property. dependsOn is a string array of artifact names that the particular artifact needs to be created before it's created.

### Note

When creating blueprint objects, each artifact resource gets its name from the filename, if using PowerShell, or the URL endpoint, if using REST API. resourceGroup references in artifacts must match those defined in the blueprint definition.

## Example - ordered resource group

This example blueprint definition has a resource group that has defined a custom sequencing order by declaring a value for dependsOn, along with a standard resource group. In this case, the artifact named assignPolicyTags will be processed before the ordered-rg resource group. standard-rg will be processed per the default sequencing order.

```
{  
  "properties": {  
    "description": "Example blueprint with custom sequencing order",  
    "resourceGroups": {  
      "ordered-rg": {  
        "dependsOn": [  
          "assignPolicyTags"  
        ],  
        "metadata": {  
          "description": "Resource Group that waits for 'assignPolicyTags' creation"  
        }  
      },  
      "standard-rg": {  
        "metadata": {  
          "description": "Resource Group that follows the standard sequence ordering"  
        }  
      }  
    },  
    "targetScope": "subscription"  
  },  
  "type": "Microsoft.Blueprint/blueprints"  
}
```

## Example - artifact with custom order

This example is a policy artifact that depends on an ARM template. By default ordering, a policy artifact would be created before the ARM template. This ordering allows the policy artifact to wait for the ARM template to be created.

```
{  
  "properties": {  
    "displayName": "Assigns an identifying tag",  
    "policyDefinitionId": "/providers/Microsoft.Authorization/policyDefinitions/2a0e14a6-  
    b0a6-4fab-991a-187a4f81c498",  
    "resourceGroup": "standard-rg",  
    "dependsOn": [  
      "customTemplate"  
    ]  
  }  
}
```

```

        ],
},
"kind": "policyAssignment",
"type": "Microsoft.Blueprint/artifacts"
}

```

Example - subscription level template artifact depending on a resource group

This example is for an ARM template deployed at the subscription level to depend on a resource group. In default ordering, the subscription level artifacts would be created before any resource groups and child artifacts in those resource groups. The resource group is defined in the blueprint definition like this:

```

"resourceGroups": {
  "wait-for-me": {
    "metadata": {
      "description": "Resource Group that is deployed prior to the subscription level
template artifact"
    }
  }
}

```

The subscription level template artifact depending on the wait-for-me resource group is defined like this:

```

{
  "properties": {
    "template": {
      ...
    },
    "parameters": {
      ...
    },
    "dependsOn": ["wait-for-me"],
    "displayName": "SubLevelTemplate",
    "description": ""
  },
  "kind": "template",
  "type": "Microsoft.Blueprint/blueprints/artifacts"
}

```

## Processing the customized sequence

During the creation process, a topological sort is used to create the dependency graph of the blueprints artifacts. The check makes sure each level of dependency between resource groups and artifacts is supported.

If an artifact dependency is declared that wouldn't alter the default order, then no change is made. An example is a resource group that depends on a subscription level policy. Another example is a resource group 'standard-rg' child policy assignment that depends on resource group 'standard-rg' child role assignment. In both cases, the dependsOn wouldn't have altered the default sequencing order and no changes would be made.

## Understand resource locking in Azure Blueprints

The creation of consistent environments at scale is only truly valuable if there's a mechanism to maintain that consistency. This article explains how resource locking works in Azure Blueprints.

**Note:** Resource locks deployed by Azure Blueprints are only applied to non-extension resources deployed by the blueprint assignment. Existing resources, such as those in resource groups that already exist, don't have locks added to them.

### Locking modes and states

Locking Mode applies to the blueprint assignment and it has three options: Don't Lock, Read Only, or Do Not Delete. The locking mode is configured during artifact deployment during a blueprint assignment. A different locking mode can be set by updating the blueprint assignment. Locking modes, however, can't be changed outside of Azure Blueprints.

Resources created by artifacts in a blueprint assignment have four states: Not Locked, Read Only, Cannot Edit / Delete, or Cannot Delete. Each artifact type can be in the Not Locked state. The following table can be used to determine the state of a resource:

Mode	Artifact Resource Type	State	Description
------	------------------------------	-------	-------------

Don't Lock	*	Not Locked	Resources aren't protected by Azure Blueprints. This state is also used for resources added to a Read Only or Do Not Delete resource group artifact from outside a blueprint assignment.
------------	---	------------	--

Read Only	Resource group	Cannot Edit / Delete	The resource group is read only and tags on the resource group can't be modified. Not Locked resources can be added, moved, changed, or deleted from this resource group.
Read Only	Non-resource group	Read Only	The resource can't be altered in any way. No changes and it can't be deleted.
Do Not Delete	*	Cannot Delete	The resources can be altered, but can't be deleted. Not Locked resources can be added, moved, changed, or deleted from this resource group.

## Overriding locking states

It's typically possible for someone with appropriate Azure role-based access control (Azure RBAC) on the subscription, such as the 'Owner' role, to be allowed to alter or delete any resource. This access isn't the case when Azure Blueprints applies locking as part of a deployed assignment. If the assignment was set with the Read Only or Do Not Delete option, not even the subscription owner can perform the blocked action on the protected resource. This security measure protects the consistency of the defined blueprint and the environment it was designed to create from accidental or programmatic deletion or alteration.

## Assign at management group

The only option to prevent subscription owners from removing a blueprint assignment is to assign the blueprint to a management group. In this scenario, only Owners of the management group have the permissions needed to remove the blueprint assignment.

To assign the blueprint to a management group instead of a subscription, the REST API call changes to look like this:

```
PUT https://management.azure.com/providers/Microsoft.Management/managementGroups/{assignmentMG}/providers/Microsoft.Blueprint/blueprintAssignments/{assignmentName}?api-version=2018-11-01-preview
```

The management group defined by {assignmentMG} must be either within the management group hierarchy or be the same management group where the blueprint definition is saved.

The request body of the blueprint assignment looks like this:

```
{
  "identity": {
    "type": "SystemAssigned"
  },
  "location": "eastus",
  "properties": {
    "description": "enforce pre-defined simpleBlueprint to this XXXXXXXX subscription.",
    "blueprintId": "/providers/Microsoft.Management/managementGroups/{blueprintMG}/providers/Microsoft.Blueprint/blueprints/simpleBlueprint",
    "scope": "/subscriptions/{targetSubscriptionId}",
    "parameters": {
      "storageAccountType": {
        "value": "Standard_LRS"
      },
      "costCenter": {
        "value": "Contoso/Online/Shopping/Production"
      },
      "owners": {
        "value": [
          "johnDoe@contoso.com",
          "johnsteam@contoso.com"
        ]
      }
    },
    "resourceGroups": {
      "storageRG": {
        "name": "defaultRG",
        "location": "eastus"
      }
    }
  }
}
```

The key difference in this request body and one being assigned to a subscription is the properties.scope property. This required property must be set to the subscription that the blueprint assignment applies to. The subscription must be a direct child of the management group hierarchy where the blueprint assignment is stored.

**Note:** A blueprint assigned to management group scope still operates as a subscription level blueprint assignment. The only difference is where the blueprint assignment is stored to prevent subscription owners from removing the assignment and associated locks.

## Removing locking states

If it becomes necessary to modify or delete a resource protected by an assignment, there are two ways to do so.

- Updating the blueprint assignment to a locking mode of Don't Lock
- Delete the blueprint assignment

When the assignment is removed, the locks created by Azure Blueprints are removed. However, the resource is left behind and would need to be deleted through normal means.

## How blueprint locks work

An Azure RBAC deny assignment deny action is applied to artifact resources during assignment of a blueprint if the assignment selected the Read Only or Do Not Delete option. The deny action is added by the managed identity of the blueprint assignment and can only be removed from the artifact resources by the same managed identity. This security measure enforces the locking mechanism and prevents removing the blueprint lock outside Azure Blueprints.

NAME	DENIED	EXCLUDED PRINCIPALS	SCOPE
Deny assignment '27ae2c52-ef23...'	All principals	Yes	This resource

The deny assignment properties of each mode are as follows:

Mode	Permissions.Action	Permissions.NotActions	PrincipalType	ExcludePrincipals[i].Id	DoNotApplyToChildScopes
Read Only	*	*/read Microsoft.Authorization/locks/delete Microsoft.Network/virtualNetwork/subnets/join/action	System Defined (Everyone)	blueprint assignment and user-defined in excludedPrincipals	Resource group - true; Resource - false
Do Not Delete	*/delete	Microsoft.Authorization/locks/delete Microsoft.Network/virtualNetwork/subnets/join/action	System Defined (Everyone)	blueprint assignment and user-defined in excludedPrincipals	Resource group - true; Resource - false

**Important:** Azure Resource Manager caches role assignment details for up to 30 minutes. As a result, deny assignments deny action's on blueprint resources may not immediately be in full effect. During this period of time, it might be possible to delete a resource intended to be protected by blueprint locks.

## Exclude a principal from a deny assignment

In some design or security scenarios, it may be necessary to exclude a principal from the deny assignment the blueprint assignment creates. This step is done in the REST API by adding up to five values to the excludedPrincipals array in the locks property when creating the assignment. The following assignment definition is an example of a request body that includes excludedPrincipals:

JSON

Copy

```
{  
  "identity": {  
    "type": "SystemAssigned"  
  },  
  "location": "eastus",  
  "properties": {  
    "description": "enforce pre-defined simpleBlueprint to this XXXXXXXX subscription.",  
    "blueprintId": "/providers/Microsoft.Management/managementGroups/{mgId}/providers/  
Microsoft.Blueprint/blueprints/simpleBlueprint",  
    "locks": {  
      "mode": "AllResourcesDoNotDelete",  
      "excludedPrincipals": [  
        "7be2f100-3af5-4c15-bcb7-27ee43784a1f",  
        "38833b56-194d-420b-90ce-cff578296714"  
      ]  
    },  
    "parameters": {  
      "storageAccountType": {  
        "value": "Standard_LRS"  
      },  
      "costCenter": {  
        "value": "Contoso/Online/Shopping/Production"  
      },  
      "owners": {  
        "value": [  
          "johnDoe@contoso.com",  
          "johnsteam@contoso.com"  
        ]  
      }  
    },  
    "resourceGroups": {  
      "storageRG": {  
        "name": "defaultRG",  
        "location": "eastus"  
      }  
    }  
}
```

```
    }
}
}
```

## Exclude an action from a deny assignment

Similar to excluding a principal on a deny assignment in a blueprint assignment, you can exclude specific Azure resource provider operations. Within the properties.locks block, in the same place that excludedPrincipals is, an excludedActions can be added:

JSON

Copy

```
"locks": {
  "mode": "AllResourcesDoNotDelete",
  "excludedPrincipals": [
    "7be2f100-3af5-4c15-bcb7-27ee43784a1f",
    "38833b56-194d-420b-90ce-cff578296714"
  ],
  "excludedActions": [
    "Microsoft.ContainerRegistry/registries/push/write",
    "Microsoft.Authorization/*/read"
  ]
},
```

While excludedPrincipals must be explicit, excludedActions entries can make use of \* for wildcard matching of resource provider operations.

## Azure Lighthouse

### What is Azure Lighthouse?

Azure Lighthouse enables multi-tenant management with scalability, higher automation, and enhanced governance across resources.

With Azure Lighthouse, service providers can deliver managed services using comprehensive and robust tooling built into the Azure platform. Customers maintain control over who has access to their tenant, which resources they can access, and what actions can be taken. Enterprise organizations managing resources across multiple tenants can also use Azure Lighthouse to streamline management tasks.

Cross-tenant management experiences lets you work more efficiently with Azure services like Azure Policy, Microsoft Sentinel, Azure Arc, and many more. Users can see

what changes were made and by whom in the activity log, which is stored in the customer's tenant and can be viewed by users in the managing tenant.



## Benefits

Azure Lighthouse helps service providers efficiently build and deliver managed services. Benefits include:

- Management at scale: Customer engagement and life-cycle operations to manage customer resources are easier and more scalable. Existing APIs, management tools, and workflows can be used with delegated resources, including machines hosted outside of Azure, regardless of the regions in which they're located.
- Greater visibility and control for customers: Customers have precise control over the scopes they delegate for management and the permissions that are allowed. They can audit service provider actions and remove access completely at any time.
- Comprehensive and unified platform tooling: Azure Lighthouse works with existing tools and APIs, Azure managed applications, and partner programs like the Cloud Solution Provider program (CSP). This flexibility supports key service provider scenarios, including multiple licensing models such as EA, CSP and pay-as-you-go. You can integrate Azure Lighthouse into your existing workflows and applications, and track your impact on customer engagements by linking your partner ID.

## Capabilities

Azure Lighthouse includes multiple ways to help streamline engagement and management:

- Azure delegated resource management: Manage your customers' Azure resources securely from within your own tenant, without having to switch context and control planes. Customer subscriptions and resource groups can be delegated to specific users and roles in the managing tenant, with the ability to remove access as needed.
- New Azure portal experiences: View cross-tenant information in the My customers page in the Azure portal. A corresponding Service providers page lets customers view and manage their service provider access.

- Azure Resource Manager templates: Use ARM templates to onboard delegated customer resources and perform cross-tenant management tasks.
- Managed Service offers in Azure Marketplace: Offer your services to customers through private or public offers, and automatically onboard them to Azure Lighthouse.

## Pricing and availability

There are no additional costs associated with using Azure Lighthouse to manage Azure resources. Any Azure customer or partner can use Azure Lighthouse.

## Cross-region and cloud considerations

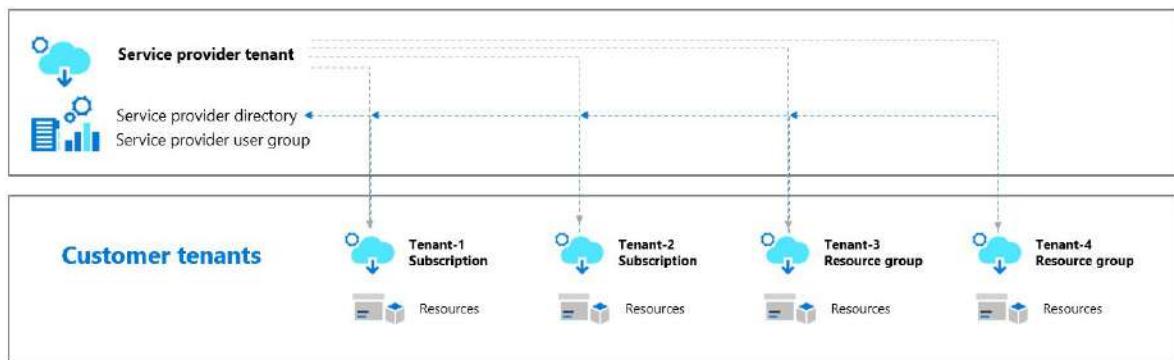
Azure Lighthouse is a non-regional service. You can manage delegated resources that are located in different regions. However, delegation of subscriptions across a national cloud and the Azure public cloud, or across two separate national clouds, isn't supported.

## Support for Azure Lighthouse

For help using Azure Lighthouse, open a support request in the Azure portal. For Issue type, choose Technical. Select a subscription, then select Lighthouse (under Monitoring & Management).

## Azure Lighthouse architecture

Azure Lighthouse helps service providers simplify customer engagement and onboarding experiences, while managing delegated resources at scale with agility and precision. Authorized users, groups, and service principals can work directly in the context of a customer subscription without having an account in that customer's Azure Active Directory (Azure AD) tenant or being a co-owner of the customer's tenant. The mechanism used to support this access is called Azure delegated resource management.



Azure Lighthouse can also be used within an enterprise which has multiple Azure AD tenants of its own to simplify cross-tenant management.

This topic discusses the relationship between tenants in Azure Lighthouse, and the resources created in the customer's tenant that enable that relationship.

## Delegation resources created in the customer tenant

When a customer's subscription or resource group is onboarded to Azure Lighthouse, two resources are created: the registration definition and the registration assignment. You can use APIs and management tools to access these resources, or work with them in the Azure portal.

### Registration definition

The registration definition contains the details of the Azure Lighthouse offer (the managing tenant ID and the authorizations that assign built-in roles to specific users, groups, and/or service principals in the managing tenant).

A registration definition is created at the subscription level for each delegated subscription, or in each subscription that contains a delegated resource group. When using APIs to create a registration definition, you'll need to work at the subscription level. For instance, using Azure PowerShell, you'll need to use `New-AzureRmDeployment` before you create a new registration definition (`New-AzManagedServicesDefinition`), rather than using `New-AzureRmResourceGroupDeployment`.

### Registration assignment

The registration assignment assigns the registration definition to a specific scope—that is, the onboarded subscription(s) and/or resource group(s).

A registration assignment is created in each delegated scope, so it will either be at the subscription group level or the resource group level, depending on what was onboarded. Each registration assignment must reference a valid registration definition at the subscription level, tying the authorizations for that service provider to the delegated scope and thus granting access.

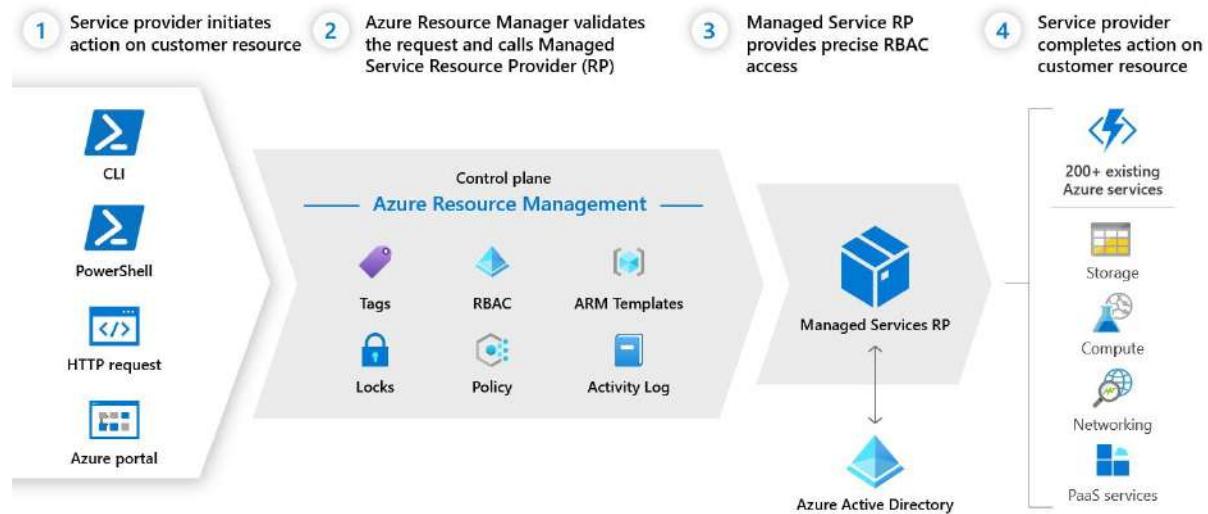
### Logical projection

Azure Lighthouse creates a logical projection of resources from one tenant onto another tenant. This lets authorized service provider users sign in to their own tenant with authorization to work in delegated customer subscriptions and resource groups. Users in the service provider's tenant can then perform management operations on behalf of their customers, without having to sign in to each individual customer tenant.

Whenever a user, group, or service principal in the service provider tenant accesses resources in a customer's tenant, Azure Resource Manager receives a request. Resource Manager authenticates these requests, just as it does for requests made by users within the customer's own tenant. For Azure Lighthouse, it does this by confirming that two resources—

the registration definition and the registration assignment—are present in the customer's tenant. If so, the Resource Manager authorizes the access according to the information defined by those resources.

## Azure delegated resource management creates logical (control plane) access to customer's environment for the service provider



Activity from users in the service provider's tenant is tracked in the activity log, which is stored in the customer's tenant. This allows the customer to see what changes were made and by whom.

## How Azure Lighthouse works

At a high level, here's how Azure Lighthouse works for the managing tenant:

1. Identify the roles that your groups, service principals, or users will need to manage the customer's Azure resources.
2. Specify this access and onboard the customer to Azure Lighthouse either by publishing a Managed Service offer to Azure Marketplace, or by deploying an Azure Resource Manager template. This onboarding process creates the two resources described above (registration definition and registration assignment) in the customer's tenant.
3. Once the customer has been onboarded, authorized users sign in to your managing tenant and perform tasks at the specified customer scope (subscription or resource group) per the access that you defined. Customers can review all actions taken, and they can remove access at any time.

While in most cases only one service provider will be managing specific resources for a customer, it's possible for the customer to create multiple delegations for the same subscription or resource group, allowing multiple service providers to have access. This scenario also enables ISV scenarios that project resources from the service provider's tenant to multiple customers.

## Cross-tenant management experiences

As a service provider, you can use Azure Lighthouse to manage resources for multiple customers from within your own Azure Active Directory (Azure AD) tenant. Many tasks and services can be performed across managed tenants by using Azure delegated resource management.

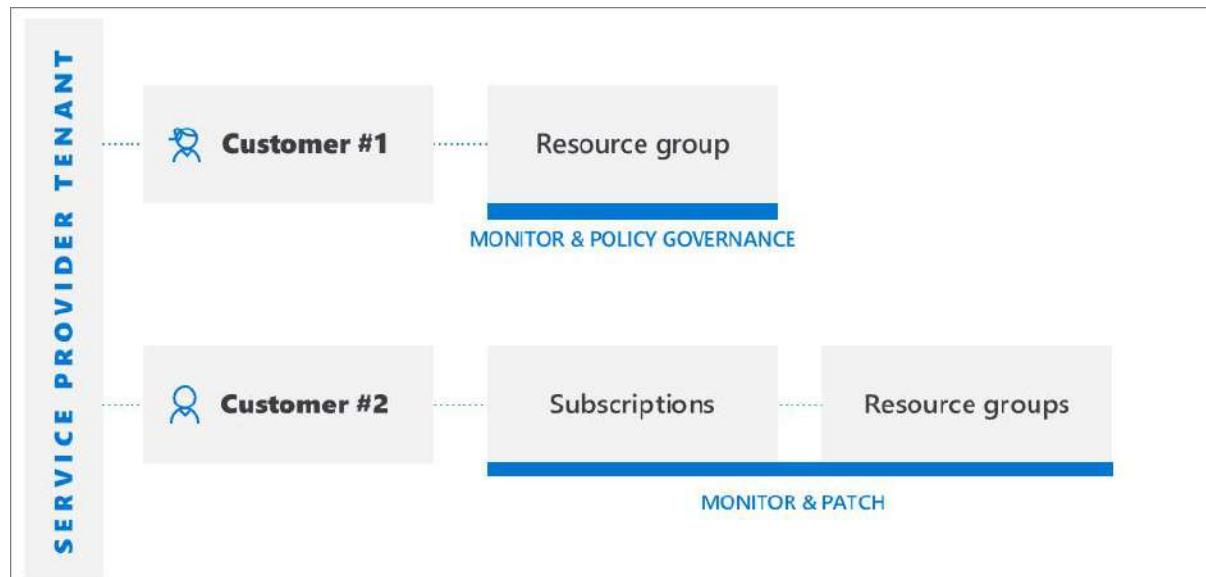
### Understanding tenants and delegation

An Azure AD tenant is a representation of an organization. It's a dedicated instance of Azure AD that an organization receives when they create a relationship with Microsoft by signing up for Azure, Microsoft 365, or other services. Each Azure AD tenant is distinct and separate from other Azure AD tenants, and has its own tenant ID (a GUID). For more info, see [What is Azure Active Directory?](#)

Typically, in order to manage Azure resources for a customer, service providers would have to sign in to the Azure portal using an account associated with that customer's tenant, requiring an administrator in the customer's tenant to create and manage user accounts for the service provider.

With Azure Lighthouse, the onboarding process specifies users within the service provider's tenant who will be able to work on delegated subscriptions and resource groups in the customer's tenant. These users can then sign in to the Azure portal using their own credentials. Within the Azure portal, they can manage resources belonging to all customers to which they have access. This can be done by visiting the [My customers](#) page in the Azure portal, or by working directly within the context of that customer's subscription, either in the Azure portal or via APIs.

Azure Lighthouse allows greater flexibility to manage resources for multiple customers without having to sign in to different accounts in different tenants. For example, a service provider may have two customers with different responsibilities and access levels. Using Azure Lighthouse, authorized users can sign in to the service provider's tenant to access these resources.



## APIs and management tool support

You can perform management tasks on delegated resources either directly in the portal or by using APIs and management tools (such as Azure CLI and Azure PowerShell). All existing APIs can be used when working with delegated resources, as long as the functionality is supported for cross-tenant management and the user has the appropriate permissions.

The Azure PowerShell Get-AzSubscription cmdlet will show the TenantId for the managing tenant by default. You can use the HomeTenantId and ManagedByTenantIds attributes for each subscription, allowing you to identify whether a returned subscription belongs to a managed tenant or to your managing tenant.

Similarly, Azure CLI commands such as az account list show the homeTenantId and managedByTenants attributes. If you don't see these values when using Azure CLI, try clearing your cache by running az account clear followed by az login —identity. In the Azure REST API, the Subscriptions - Get and Subscriptions - List commands include ManagedByTenant.

**Note:** In addition to tenant information related to Azure Lighthouse, tenants shown by these APIs may also reflect partner tenants for Azure Databricks or Azure managed applications.

## Enhanced services and scenarios

Most tasks and services can be performed on delegated resources across managed tenants. Below are some of the key scenarios where cross-tenant management can be especially effective.

Azure Arc:

- Manage hybrid servers at scale - Azure Arc-enabled servers:
  - Manage Windows Server or Linux machines outside Azure that are connected to delegated subscriptions and/or resource groups in Azure
  - Manage connected machines using Azure constructs, such as Azure Policy and tagging
  - Ensure the same set of policies are applied across customers' hybrid environments
  - Use Microsoft Defender for Cloud to monitor compliance across customers' hybrid environments
- Manage hybrid Kubernetes clusters at scale - Azure Arc-enabled Kubernetes:
  - Manage Kubernetes clusters that are connected to delegated subscriptions and/or resource groups in Azure
  - Use GitOps for connected clusters
  - Enforce policies across connected clusters

Azure Automation:

- Use Automation accounts to access and work with delegated resources

Azure Backup:

- Backup and restore customer data from on-premises workloads, Azure VMs, Azure file shares, and more

- View data for all delegated customer resources in Backup Center
- Use the Backup Explorer to help view operational information of backup items (including Azure resources not yet configured for backup) and monitoring information (jobs and alerts) for delegated subscriptions. The Backup Explorer is currently available only for Azure VM data.
- Use Backup Reports across delegated subscriptions to track historical trends, analyze backup storage consumption, and audit backups and restores.

#### Azure Blueprints:

- Use Azure Blueprints to orchestrate the deployment of resource templates and other artifacts (requires additional access to prepare the customer subscription)

#### Azure Cost Management + Billing:

- From the managing tenant, CSP partners can view, manage, and analyze pre-tax consumption costs (not inclusive of purchases) for customers who are under the Azure plan. The cost will be based on retail rates and the Azure role-based access control (Azure RBAC) access that the partner has for the customer's subscription. Currently, you can view consumption costs at retail rates for each individual customer subscription based on Azure RBAC access.

#### Azure Key Vault:

- Create Key Vaults in customer tenants
- Use a managed identity to create Key Vaults in customer tenants

#### Azure Kubernetes Service (AKS):

- Manage hosted Kubernetes environments and deploy and manage containerized applications within customer tenants
- Deploy and manage clusters in customer tenants
- Use Azure Monitor for containers to monitor performance across customer tenants

#### Azure Migrate:

- Create migration projects in the customer tenant and migrate VMs

#### Azure Monitor:

- View alerts for delegated subscriptions, with the ability to view and refresh alerts across all subscriptions
- View activity log details for delegated subscriptions
- Log analytics: Query data from remote workspaces in multiple tenants (note that automation accounts used to access data from workspaces in customer tenants must be created in the same tenant)
- Create, view, and manage metric alerts, log alerts, and activity log alerts in customer tenants
- Create alerts in customer tenants that trigger automation, such as Azure Automation runbooks or Azure Functions, in the managing tenant through webhooks
- Create diagnostic settings in workspaces created in customer tenants, to send resource logs to workspaces in the managing tenant
- For SAP workloads, monitor SAP Solutions metrics with an aggregated view across customer tenants
- For Azure AD B2C, route sign-in and auditing logs to different monitoring solutions

#### Azure Networking:

- Deploy and manage Azure Virtual Network and virtual network interface cards (vNICs) within managed tenants
- Deploy and configure Azure Firewall to protect customers' Virtual Network resources
- Manage connectivity services such as Azure Virtual WAN, ExpressRoute, and VPN Gateways
- Use Azure Lighthouse to support key scenarios for the Azure Networking MSP Program

#### Azure Policy:

- Create and edit policy definitions within delegated subscriptions
- Deploy policy definitions and policy assignments across multiple tenants
- Assign customer-defined policy definitions within delegated subscriptions
- Customers see policies authored by the service provider alongside any policies they've authored themselves
- Can remediate deployIfNotExists or modify assignments within the managed tenant
- Note that viewing compliance details for non-compliant resources in customer tenants is not currently supported

#### Azure Resource Graph:

- Now includes the tenant ID in returned query results, allowing you to identify whether a subscription belongs to a managed tenant

#### Azure Service Health:

- Monitor the health of customer resources with Azure Resource Health
- Track the health of the Azure services used by your customers

#### Azure Site Recovery:

- Manage disaster recovery options for Azure virtual machines in customer tenants (note that you can't use RunAs accounts to copy VM extensions)

#### Azure Virtual Machines:

- Use virtual machine extensions to provide post-deployment configuration and automation tasks on Azure VMs
- Use boot diagnostics to troubleshoot Azure VMs
- Access VMs with serial console
- Integrate VMs with Azure Key Vault for passwords, secrets, or cryptographic keys for disk encryption by using managed identity through policy, ensuring that secrets are stored in a Key Vault in the managed tenants
- Note that you can't use Azure Active Directory for remote login to VMs

#### Microsoft Defender for Cloud:

- Cross-tenant visibility
  - Monitor compliance to security policies and ensure security coverage across all tenants' resources
  - Continuous regulatory compliance monitoring across multiple tenants in a single view
  - Monitor, triage, and prioritize actionable security recommendations with secure score calculation

- Cross-tenant security posture management
  - Manage security policies
  - Take action on resources that are out of compliance with actionable security recommendations
  - Collect and store security-related data
- Cross-tenant threat detection and protection
  - Detect threats across tenants' resources
  - Apply advanced threat protection controls such as just-in-time (JIT) VM access
  - Harden network security group configuration with Adaptive Network Hardening
  - Ensure servers are running only the applications and processes they should be with adaptive application controls
  - Monitor changes to important files and registry entries with File Integrity Monitoring (FIM)
- Note that the entire subscription must be delegated to the managing tenant; Microsoft Defender for Cloud scenarios are not supported with delegated resource groups

Microsoft Sentinel:

- Manage Microsoft Sentinel resources in customer tenants
- Track attacks and view security alerts across multiple tenants
- View incidents across multiple Microsoft Sentinel workspaces spread across tenants

Support requests:

- Open support requests from Help + support in the Azure portal for delegated resources (selecting the support plan available to the delegated scope)
- Use the Azure Quota API to view and manage Azure service quotas for delegated customer resources

## Current limitations

With all scenarios, please be aware of the following current limitations:

- Requests handled by Azure Resource Manager can be performed using Azure Lighthouse. The operation URIs for these requests start with <https://management.azure.com>. However, requests that are handled by an instance of a resource type (such as Key Vault secrets access or storage data access) aren't supported with Azure Lighthouse. The operation URIs for these requests typically start with an address that is unique to your instance, such as <https://myaccount.blob.core.windows.net> or <https://mykeyvault.vault.azure.net/>. The latter also are typically data operations rather than management operations.
- Role assignments must use Azure built-in roles. All built-in roles are currently supported with Azure Lighthouse, except for Owner or any built-in roles with DataActions permission. The User Access Administrator role is supported only for limited use in assigning roles to managed identities. Custom roles and classic subscription administrator roles are not supported.

- While you can onboard subscriptions that use Azure Databricks, users in the managing tenant can't launch Azure Databricks workspaces on a delegated subscription at this time.
- While you can onboard subscriptions and resource groups that have resource locks, those locks will not prevent actions from being performed by users in the managing tenant. Deny assignments that protect system-managed resources, such as those created by Azure managed applications or Azure Blueprints (system-assigned deny assignments), do prevent users in the managing tenant from acting on those resources; however, at this time users in the customer tenant can't create their own deny assignments (user-assigned deny assignments).
- Delegation of subscriptions across a national cloud and the Azure public cloud, or across two separate national clouds, is not supported.

## Tenants, users, and roles in Azure Lighthouse scenarios

Before onboarding customers for Azure Lighthouse, it's important to understand how Azure Active Directory (Azure AD) tenants, users, and roles work, and how they can be used in Azure Lighthouse scenarios.

A tenant is a dedicated and trusted instance of Azure AD. Typically, each tenant represents a single organization. Azure Lighthouse enables logical projection of resources from one tenant to another tenant. This allows users in the managing tenant (such as one belonging to a service provider) to access delegated resources in a customer's tenant, or lets enterprises with multiple tenants centralize their management operations.

In order to achieve this logical projection, a subscription (or one or more resource groups within a subscription) in the customer tenant must be onboarded to Azure Lighthouse. This onboarding process can be done either through Azure Resource Manager templates or by publishing a public or private offer to Azure Marketplace.

With either onboarding method, you'll need to define authorizations. Each authorization includes a principalId (an Azure AD user, group, or service principal in the managing tenant) combined with a built-in role that defines the specific permissions that will be granted for the delegated resources.

**Note:** Unless explicitly specified, references to a "user" in the Azure Lighthouse documentation can apply to an Azure AD user, group, or service principal in an authorization.

## Best practices for defining users and roles

When creating your authorizations, we recommend the following best practices:

- In most cases, you'll want to assign permissions to an Azure AD user group or service principal, rather than to a series of individual user accounts. This lets you add or remove access for individual users through your tenant's Azure AD, rather than having to update the delegation every time your individual access requirements change.

- Follow the principle of least privilege so that users only have the permissions needed to complete their job, helping to reduce the chance of inadvertent errors.
- Include an authorization with the Managed Services Registration Assignment Delete Role so that you can remove access to the delegation later if needed. If this role is not assigned, access to delegated resources can only be removed by a user in the customer's tenant.
- Be sure that any user who needs to view the My customers page in the Azure portal has the Reader role (or another built-in role which includes Reader access).

In order to add permissions for an Azure AD group, the Group type must be set to Security. This option is selected when the group is created.

## Role support for Azure Lighthouse

When defining an authorization, each user account must be assigned one of the Azure built-in roles. Custom roles and classic subscription administrator roles are not supported.

All built-in roles are currently supported with Azure Lighthouse, with the following exceptions:

- The Owner role is not supported.
- Any built-in roles with DataActions permission are not supported.
- The User Access Administrator built-in role is supported, but only for the limited purpose of assigning roles to a managed identity in the customer tenant. No other permissions typically granted by this role will apply. If you define a user with this role, you must also specify the built-in role(s) that this user can assign to managed identities.

**Note:** Once a new applicable built-in role is added to Azure, it can be assigned when onboarding a customer using Azure Resource Manager templates. There may be a delay before the newly-added role becomes available in Partner Center when publishing a managed service offer.

## Transferring delegated subscriptions between Azure AD tenants

If a subscription is transferred to another Azure AD tenant account, the registration definition and registration assignment resources created through the Azure Lighthouse onboarding process will be preserved. This means that access granted through Azure Lighthouse to managing tenants will remain in effect for that subscription (or for delegated resource groups within that subscription).

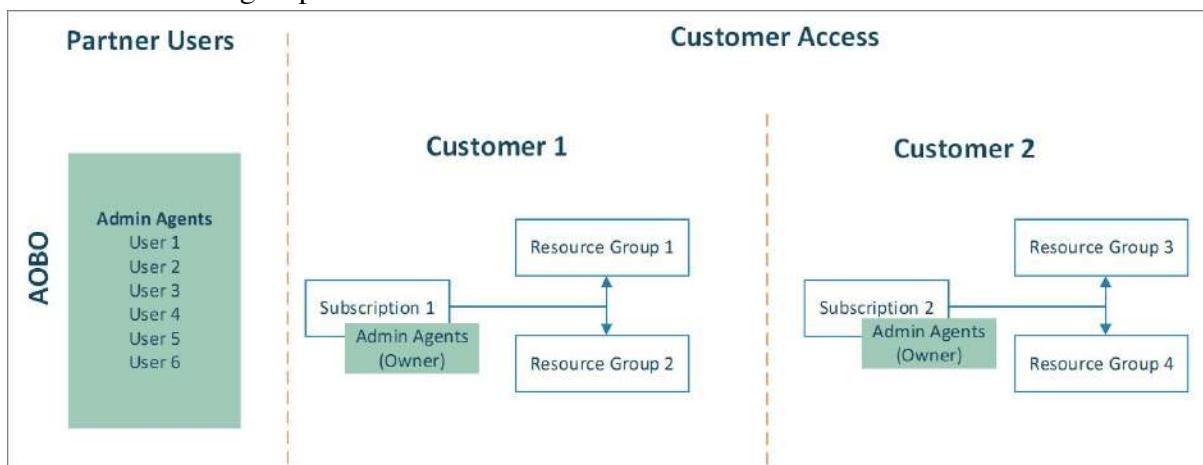
The only exception is if the subscription is transferred to an Azure AD tenant to which it had been previously delegated. In this case, the delegation resources for that tenant are removed and the access granted through Azure Lighthouse will no longer apply, since the subscription now belongs directly to that tenant (rather than being delegated to it through Azure Lighthouse). However, if that subscription had also been delegated to other managing tenants, those other managing tenants will retain the same access to the subscription.

# Azure Lighthouse and the Cloud Solution Provider program

With Azure Lighthouse, you can use Azure delegated resource management along with AOBO. This helps improve security and reduces unnecessary access by enabling more granular permissions for your users. It also allows for greater efficiency and scalability, as your users can work across multiple customer subscriptions using a single login in your tenant.

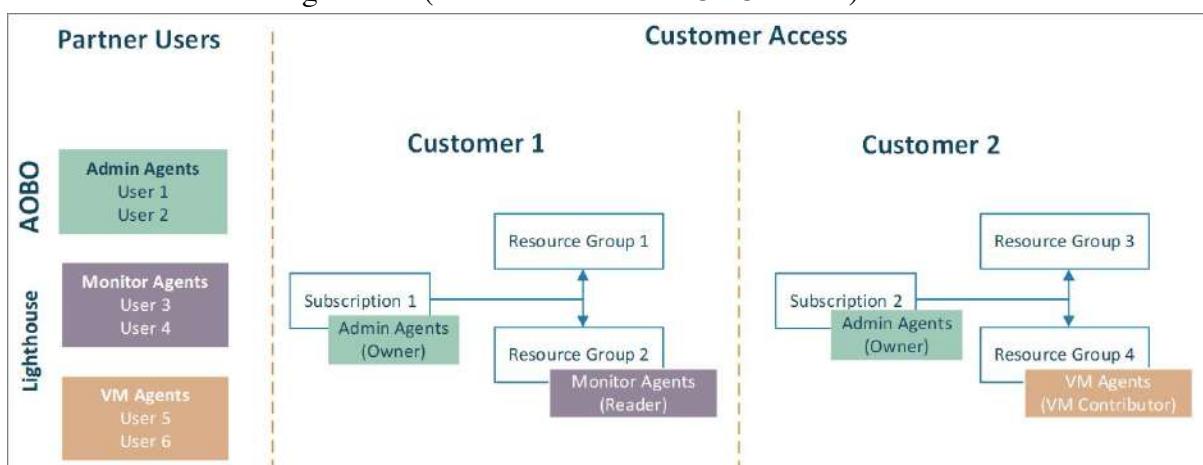
## Administer on Behalf of (AOBO)

With AOBO, any user with the Admin Agent role in your tenant will have AOBO access to Azure subscriptions that you create through the CSP program. Any users who need access to any customers' subscriptions must be a member of this group. AOBO doesn't allow the flexibility to create distinct groups that work with different customers, or to enable different roles for groups or users.



## Azure Lighthouse

Using Azure Lighthouse, you can assign different groups to different customers or roles, as shown in the following diagram. Because users will have the appropriate level of access through Azure delegated resource management, you can reduce the number of users who have the Admin Agent role (and thus have full AOBO access).



Azure Lighthouse helps improve security by limiting unnecessary access to your customers' resources. It also gives you more flexibility to manage multiple customers at scale, using the Azure built-in role that's most appropriate for each user's duties, without granting a user more access than necessary.

To further minimize the number of permanent assignments, you can create eligible authorizations (currently in public preview) to grant additional permissions to your users on a just-in-time basis.

Onboarding a subscription that you created through the CSP program follows the steps described in Onboard a subscription to Azure Lighthouse. Any user who has the Admin Agent role in your tenant can perform this onboarding.

## Azure Lighthouse and Azure managed applications

Both Azure managed applications and Azure Lighthouse work by enabling a service provider to access resources that reside in the customer's tenant. It can be helpful to understand the differences in the way that they work and the scenarios that they help to enable, and how they can be used together.

### Comparing Azure Lighthouse and Azure managed applications

This table illustrates some high-level differences that may impact whether you might choose to use Azure Lighthouse or Azure managed applications. As noted below, you can also design a solution that uses them together.

Consideration	Azure Lighthouse	Azure managed applications
Typical user	Service providers or enterprises managing multiple tenants	Independent Software Vendors (ISVs)
Scope of cross-tenant access	Subscription(s) or resource group(s)	Resource group (scoped to a single application)
Purchasable in Azure Marketplace	No (offers can be published to Azure Marketplace, but customers are billed separately)	Yes

IP protection	Yes (IP can remain in the service provider's tenant)	Yes (by design, resource group is locked to customers)
Deny assignments	No	Yes

## Azure Lighthouse

With Azure Lighthouse, a service provider can perform a wide range of management tasks directly on a customer's subscription (or resource group). This access is achieved through a logical projection, allowing service providers to sign in to their own tenant and access resources that belong to the customer's tenant. The customer can determine which subscriptions or resource groups to delegate to the service provider, and the customer maintains full access to those resources. They can also remove the service provider's access at any time.

To use Azure Lighthouse, customers are onboarded either by deploying ARM templates or through a Managed Service offer in Azure Marketplace. You can track your impact on customer engagements by linking your partner ID.

Azure Lighthouse is typically used when a service provider will perform management tasks for a customer on an ongoing basis.

## Azure managed applications

Azure managed applications allow a service provider or ISV to offer cloud solutions that are easy for customers to deploy and use in their own subscriptions.

In a managed application, the resources used by the application are bundled together and deployed to a resource group that's managed by the publisher. This resource group is present in the customer's subscription, but an identity in the publisher's tenant has access to it. The ISV continues to manage and maintain the managed application, while the customer does not have direct access to work in its resource group, or any access to its resources.

Managed applications support customized Azure portal experiences and integration with custom providers. These options can be used to deliver a more customized and integrated experience, making it easier for customers to perform some management tasks themselves.

Managed applications can be published to Azure Marketplace, either as a private offer for a specific customer's use, or as public offers that multiple customers can purchase. They can also be delivered to users within your organization by publishing managed applications to your service catalog. You can deploy both service catalog and Marketplace instances using ARM templates, which can include a Commercial Marketplace partner's unique identifier to track customer usage attribution.

Azure managed applications are typically used for a specific customer need that can be achieved through a turnkey solution that is fully managed by the service provider.

## Using Azure Lighthouse and Azure managed applications together

While Azure Lighthouse and Azure managed applications use different access mechanisms to achieve different goals, there may be scenarios where it makes sense for a service provider to use both of them with the same customer.

For example, a customer might want managed services delivered by a service provider through Azure Lighthouse, so that they have visibility into the partner's actions along with continued control of their delegated subscription. However, the service provider may not want the customer to access certain resources that will be stored in the customer's tenant, or allow any customized actions on those resources. To meet these goals, the service provider can publish a private offer as a managed application. The managed application can include a resource group that is deployed in the customer's tenant, but that can't be accessed directly by the customer.

Customers might also be interested in managed applications from multiple service providers, whether or not they also use managed services via Azure Lighthouse from any of those service providers. Additionally, partners in the Cloud Solution Provider (CSP) program can resell certain managed applications published by other ISVs to customers that they support through Azure Lighthouse. With a wide range of options, service providers can choose the right balance to meet their customers' needs while restricting access to resources when appropriate.

## Security

### Recommended security practices

When using Azure Lighthouse, it's important to consider security and access control. Users in your tenant will have direct access to customer subscriptions and resource groups, so you'll want to take steps to maintain your tenant's security. You'll also want to make sure you only allow the access that's needed to effectively manage your customers' resources. This topic provides recommendations to help you do so.

These recommendations also apply to enterprises managing multiple tenants with Azure Lighthouse.

### Require Azure AD Multi-Factor Authentication

Azure AD Multi-Factor Authentication (also known as two-step verification) helps prevent attackers from gaining access to an account by requiring multiple authentication steps. You should require Azure AD Multi-Factor Authentication for all users in your managing tenant, including users who will have access to delegated customer resources.

It is recommended that you ask your customers to implement Azure AD Multi-Factor Authentication in their tenants as well.

## Assign permissions to groups, using the principle of least privilege

To make management easier, use Azure Active Directory (Azure AD) groups for each role required to manage your customers' resources. This lets you add or remove individual users to the group as needed, rather than assigning permissions directly to each user. In order to add permissions for an Azure AD group, the Group type must be set to Security. This option is selected when the group is created.

When creating your permission structure, be sure to follow the principle of least privilege so that users only have the permissions needed to complete their job, helping to reduce the chance of inadvertent errors.

For example, you may want to use a structure like this:

Group name	Type	principalId	Role definition	Role definition ID
Architects	User group	<principalId>	Contributor	b24988ac-6180-42a0-ab88-20f7382dd24c
Assessment	User group	<principalId>	Reader	acdd72a7-3385-48ef-bd42-f606fba81ae7
VM Specialists	User group	<principalId>	VM Contributor	9980e02c-c2be-4d73-94e8-173b1dc7cf3c
Automation	Service principal name (SPN)	<principalId>	Contributor	b24988ac-6180-42a0-ab88-20f7382dd24c

Once you've created these groups, you can assign users as needed. Only add the users who truly need to have access. Be sure to review group membership regularly and remove any users that are no longer appropriate or necessary to include.

Keep in mind that when you onboard customers through a public managed service offer, any group (or user or service principal) that you include will have the same permissions for every customer who purchases the plan. To assign different groups to work with each customer, you'll need to publish a separate private plan that is exclusive to each customer, or onboard

customers individually by using Azure Resource Manager templates. For example, you could publish a public plan that has very limited access, then work with the customer directly to onboard their resources for additional access using a customized Azure Resource Template granting additional access as needed.

You can also create eligible authorizations that let users in your managing tenant temporarily elevate their role. By using eligible authorizations, you can minimize the number of permanent assignments of users to privileged roles, helping to reduce security risks related to privileged access by users in your tenant. This feature is currently in public preview and has specific licensing requirements.

## Azure security baseline for Azure Lighthouse

This security baseline applies guidance from the Azure Security Benchmark version 2.0 to Azure Lighthouse. The Azure Security Benchmark provides recommendations on how you can secure your cloud solutions on Azure. The content is grouped by the security controls defined by the Azure Security Benchmark and the related guidance applicable to Azure Lighthouse.

**Note:** Controls not applicable to Azure Lighthouse, and those for which the global guidance is recommended verbatim, have been excluded. To see how Azure Lighthouse completely maps to the Azure Security Benchmark.

## Network Security

### NS-7: Secure Domain Name Service (DNS)

**Guidance:** Not applicable; Azure Lighthouse does not expose its underlying DNS configurations, these settings are maintained by Microsoft.

**Responsibility:** Microsoft

Microsoft Defender for Cloud monitoring: None

## Identity Management

### IM-1: Standardize Azure Active Directory as the central identity and authentication system

**Guidance:** Azure Lighthouse uses Azure Active Directory (Azure AD) as the default identity and access management service. Standardize Azure AD to govern your organization's identity and access management in:

- Microsoft Cloud resources, such as the Azure portal, Azure Storage, Azure Virtual Machine (Linux and Windows), Azure Key Vault, PaaS, and SaaS applications.
- Your organization's resources, such as applications on Azure or your corporate network resources.

With Azure Lighthouse, designated users in a managing tenant have an Azure built-in role which lets them access delegated subscriptions and/or resource groups in a customer's tenant.

All built-in roles are currently supported except for Owner or any built-in roles with DataActions permission. The User Access Administrator role is supported only for limited use in assigning roles to managed identities. Custom roles and classic subscription administrator roles are not supported.

Responsibility: Customer

Microsoft Defender for Cloud monitoring: None

#### **IM-2: Manage application identities securely and automatically**

Guidance: Azure managed identities can authenticate to Azure services and resources that support Azure Active Directory (Azure AD) authentication. Authentication is enabled through pre-defined access grant rules, avoiding hard-coded credentials in source code or configuration files. With Azure Lighthouse, users with the User Access Administrator role on a customer's subscription can create a managed identity in that customer's tenant. While this role is not generally supported with Azure Lighthouse, it can be used in this specific scenario, allowing the users with this permission to assign one or more specific built-in roles to managed identities.

For services that do not support managed identities, use Azure AD to create a service principal with restricted permissions at the resource level instead. Azure Lighthouse allows service principals to access customer resources according to the roles they are granted during the onboarding process. It is recommended to configure service principals with certificate credentials and fall back to client secrets. In both cases, Azure Key Vault can be used in conjunction with Azure managed identities, so that the runtime environment (such as an Azure function) can retrieve the credential from the key vault.

Responsibility: Customer

Microsoft Defender for Cloud monitoring: None

### **Privileged Access**

#### **PA-1: Protect and limit highly privileged users**

Guidance: Limit the number of highly privileged user accounts, and protect these accounts at an elevated level. A Global Administrator account is not required to enable and use Azure Lighthouse.

To access tenant-level Activity Log data, an account must be assigned the Monitoring Reader Azure built-in role at root scope (/). Because the Monitoring Reader role at root scope is a broad level of access, we recommend that you assign this role to a service principal account, rather than to an individual user or to a group. This assignment must be performed by a user who has the Global Administrator role with additional elevated access. This elevated access should be added immediately before making the role assignment, then removed when the assignment is complete.

Responsibility: Customer

Microsoft Defender for Cloud monitoring: None

### **PA-3: Review and reconcile user access regularly**

Guidance: Azure Lighthouse uses Azure Active Directory (Azure AD) accounts to manage its resources, review user accounts and access assignments regularly to ensure the accounts and their access are valid. You can use Azure AD access reviews to review group memberships, access to enterprise applications, and role assignments. Azure AD reporting can provide logs to help discover stale accounts. You can also use Azure AD Privileged Identity Management to create access review report workflow to facilitate the review process.

Customers can review the level of access granted to users in the managing tenant via Azure Lighthouse in the Azure portal. They can remove this access at any time.

In addition, Azure Privileged Identity Management can also be configured to alert when an excessive number of administrator accounts are created, and to identify administrator accounts that are stale or improperly configured.

Note: that some Azure services support local users and roles which are not managed through Azure AD. You will need to manage these users separately.

Responsibility: Customer

Microsoft Defender for Cloud monitoring: None

### **PA-6: Use privileged access workstations**

Guidance: Secured, isolated workstations are critically important for the security of sensitive roles like administrators, developers, and critical service operators. Depending on your requirements, you can use highly secured user workstations and/or Azure Bastion for performing administrative tasks with Azure Lighthouse in production environments. Use Azure Active Directory (Azure AD), Microsoft Defender Advanced Threat Protection (ATP), and/or Microsoft Intune to deploy a secure and managed user workstation for administrative tasks. The secured workstations can be centrally managed to enforce secured configuration, including strong authentication, software and hardware baselines, and restricted logical and network access.

Responsibility: Customer

Microsoft Defender for Cloud monitoring: None

### **PA-7: Follow just enough administration (least privilege principle)**

Guidance: Azure Lighthouse is integrated with Azure role-based access control (RBAC) to manage its resources. Azure RBAC allows you to manage Azure resource access through role assignments. You can assign these built-in roles to users, groups, service principals and managed identities. There are pre-defined built-in roles for certain resources, and these roles can be inventoried or queried through tools such as Azure CLI, Azure PowerShell or the Azure portal. The privileges you assign to resources through the Azure RBAC should be always limited to what is required by the roles. This complements the just in time (JIT) approach of Azure Active Directory (Azure AD) Privileged Identity Management (PIM) and

should be reviewed periodically. Use built-in roles to allocate permission and only create custom roles when required.

Azure Lighthouse allows access to delegated customer resources using Azure built-in roles. In most cases, you'll want to assign these roles to a group or service principal, rather than to many individual user accounts. This lets you add or remove access for individual users without having to update and republish the plan when your access requirements change.

To delegate customer resources to a managing tenant, a deployment must be done by a non-guest account in the customer's tenant who has the Owner built-in role for the subscription being onboarded (or which contains the resource groups that are being onboarded).

Responsibility: Customer

Microsoft Defender for Cloud monitoring: None

## Asset Management

### AM-1: Ensure security team has visibility into risks for assets

Guidance: Ensure security teams are granted Security Reader permissions in your Azure tenant and subscriptions so they can monitor for security risks using Microsoft Defender for Cloud.

Depending on how security team responsibilities are structured, monitoring for security risks could be the responsibility of a central security team or a local team. That said, security insights and risks must always be aggregated centrally within an organization.

Security Reader permissions can be applied broadly to an entire tenant (Root Management Group) or scoped to management groups or specific subscriptions.

Note: Additional permissions might be required to get visibility into workloads and services.

Responsibility: Customer

Microsoft Defender for Cloud monitoring: None

### AM-2: Ensure security team has access to asset inventory and metadata

Guidance: Customers' security teams can review activity logs to see activity taken by service providers who use Azure Lighthouse.

If a service provider wants to allow their security team to review delegated customer resources, the security team's authorizations should include the Reader built-in role.

Responsibility: Customer

Microsoft Defender for Cloud monitoring: None

### AM-3: Use only approved Azure services

Guidance: Use Azure Policy to audit and restrict which services users can provision in your environment. Use Azure Resource Graph to query for and discover resources within their subscriptions. You can also use Azure Monitor to create rules to trigger alerts when a non-approved service is detected.

Responsibility: Customer

Microsoft Defender for Cloud monitoring: None

## Logging and Threat Detection

### LT-1: Enable threat detection for Azure resources

Guidance: Through Azure Lighthouse, you can monitor your customers' Azure resources for potential threats and anomalies. Focus on getting high-quality alerts to reduce false positives for analysts to sort through. Alerts can be sourced from log data, agents, or other data.

Use the Microsoft Defender for Cloud built-in threat detection capability, which is based on monitoring Azure service telemetry and analyzing service logs. Data is collected using the Log Analytics agent, which reads various security-related configurations and event logs from the system and copies the data to your workspace for analysis.

In addition, use Microsoft Sentinel to build analytics rules, which hunt threats that match specific criteria across your customer's environment. The rules generate incidents when the criteria are matched, so that you can investigate each incident. Microsoft Sentinel can also import third-party threat intelligence to enhance its threat detection capability.

Responsibility: Customer

Microsoft Defender for Cloud monitoring: None

### LT-2: Enable threat detection for Azure identity and access management

Guidance: Through Azure Lighthouse, you can use Microsoft Defender for Cloud to alert on certain suspicious activities in the customer tenants you manage, such as an excessive number of failed authentication attempts, and deprecated accounts in the subscription.

Azure Active Directory (Azure AD) provides the following user logs that can be viewed in Azure AD reporting or integrated with Azure Monitor, Microsoft Sentinel or other SIEM/monitoring tools for more sophisticated monitoring and analytics use cases:

- Sign-in – The sign-in report provides information about the usage of managed applications and user sign-in activities.
- Audit logs - Provides traceability through logs for all changes done by various features within Azure AD. Examples of audit logs include changes made to any resources within Azure AD like adding or removing users, apps, groups, roles and policies.
- Risky sign-in - A risky sign-in is an indicator for a sign-in attempt that might have been performed by someone who is not the legitimate owner of a user account.
- Users flagged for risk - A risky user is an indicator for a user account that might have been compromised.

Microsoft Defender for Cloud can also alert on certain suspicious activities such as excessive number of failed authentication attempts, deprecated accounts in the subscription. In addition to the basic security hygiene monitoring, Microsoft Defender for Cloud's Threat Protection module can also collect more in-depth security alerts from individual Azure compute

resources (virtual machines, containers, app service), data resources (SQL DB and storage), and Azure service layers. This capability provides visibility on account anomalies inside the individual resources.

Responsibility: Customer

Microsoft Defender for Cloud monitoring: None

#### **LT-4: Enable logging for Azure resources**

Guidance: Activity logs, which are automatically available, contain all write operations (PUT, POST, DELETE) for your Azure Lighthouse resources except read operations (GET). Activity logs can be used to find an error when troubleshooting or to monitor how a user in your organization modified a resource.

With Azure Lighthouse, you can use Azure Monitor Logs in a scalable way across the customer tenants you're managing. Create Log Analytics workspaces directly in the customer tenants so that customer data remains in their tenants rather than being exported into yours. This also allows centralized monitoring of any resources or services supported by Log Analytics, giving you more flexibility on what types of data you monitor.

Customers who have delegated subscriptions for Azure Lighthouse can view Azure Activity log data to see all actions taken. This gives customers full visibility into operations that service providers are performing, along with operations done by users within the customer's own Azure Active Directory (Azure AD) tenant.

Responsibility: Shared

Microsoft Defender for Cloud monitoring: None

#### **LT-5: Centralize security log management and analysis**

Guidance: Centralize logging storage and analysis to enable correlation. For each log source, ensure you have assigned a data owner, access guidance, storage location, what tools are used to process and access the data, and data retention requirements.

Ensure you are integrating Azure Activity logs into your central logging. Ingest logs via Azure Monitor to aggregate security data generated by endpoint devices, network resources, and other security systems. In Azure Monitor, use Log Analytics workspaces to query and perform analytics, and use Azure Storage accounts for long term and archival storage.

In addition, enable and onboard data to Microsoft Sentinel or a third-party SIEM.

With Azure Lighthouse, you can use Azure Monitor Logs in a scalable way across the customer tenants you're managing. Create Log Analytics workspaces directly in the customer tenants so that customer data remains in their tenants rather than being exported into yours. This also allows centralized monitoring of any resources or services supported by Log Analytics, giving you more flexibility on what types of data you monitor.

Customers who have delegated subscriptions for Azure Lighthouse can view Azure Activity log data to see all actions taken. This gives customers full visibility into operations that service providers are performing, along with operations done by users within the customer's own Azure Active Directory (Azure AD) tenant.

Many organizations choose to use Microsoft Sentinel for “hot” data that is used frequently and Azure Storage for “cold” data that is used less frequently.

Responsibility: Customer

Microsoft Defender for Cloud monitoring: None

#### **LT-6: Configure log storage retention**

Guidance: Azure Lighthouse does not currently produce any security-related logs. Customers who want to view service provider activity can configure log retention according to compliance, regulation, and business requirements.

In Azure Monitor, you can set your Log Analytics workspace retention period according to your organization's compliance regulations. Use Azure Storage, Data Lake or Log Analytics workspace accounts for long-term and archival storage.

Responsibility: Customer

Microsoft Defender for Cloud monitoring: None

#### **LT-7: Use approved time synchronization sources**

Guidance: Azure Lighthouse does not support configuring your own time synchronization sources. The Azure Lighthouse service relies on Microsoft time synchronization sources, and is not exposed to customers for configuration.

Responsibility: Microsoft

Microsoft Defender for Cloud monitoring: None

### **Posture and Vulnerability Management**

#### **PV-1: Establish secure configurations for Azure services**

Guidance: Azure Lighthouse supports below service-specific policies that are available in Microsoft Defender for Cloud to audit and enforce configurations of your Azure resources. This can be configured in Microsoft Defender for Cloud or Azure Policy initiatives.

- Allow managing tenant IDs to onboard through Azure Lighthouse
- Audit delegation of scopes to a managing tenant

You can use Azure Blueprints to automate deployment and configuration of services and application environments including Azure Resource Manager templates, Azure RBAC controls, and policies, in a single blueprint definition.

Responsibility: Customer

Microsoft Defender for Cloud monitoring: None

## **PV-2: Sustain secure configurations for Azure services**

Guidance: Azure Lighthouse supports below service-specific policies that are available in Microsoft Defender for Cloud to audit and enforce configurations of your Azure resources. This can be configured in Microsoft Defender for Cloud or Azure Policy initiatives.

Responsibility: Customer

Microsoft Defender for Cloud monitoring: None

## **PV-3: Establish secure configurations for compute resources**

Guidance: Use Microsoft Defender for Cloud and Azure Policy to establish secure configurations on all compute resources including VMs, containers, and others.

Responsibility: Customer

Microsoft Defender for Cloud monitoring: None

## **PV-6: Perform software vulnerability assessments**

Guidance: Microsoft performs vulnerability management on the underlying systems that support Azure Lighthouse.

Responsibility: Microsoft

Microsoft Defender for Cloud monitoring: None

## **PV-8: Conduct regular attack simulation**

Guidance: As required, conduct penetration testing or red team activities on your Azure resources and ensure remediation of all critical security findings. Follow the Microsoft Cloud Penetration Testing Rules of Engagement to ensure your penetration tests are not in violation of Microsoft policies. Use Microsoft's strategy and execution of Red Teaming and live site penetration testing against Microsoft-managed cloud infrastructure, services, and applications.

Responsibility: Shared

Microsoft Defender for Cloud monitoring: None

## **Endpoint Security**

### **ES-1: Use Endpoint Detection and Response (EDR)**

Guidance: Azure Lighthouse does not deploy any customer-facing compute resources which would require Endpoint Detection and Response (EDR) protection. The underlying infrastructure for the Azure Lighthouse service is handled by Microsoft.

Responsibility: Microsoft

Microsoft Defender for Cloud monitoring: None

## **ES-2: Use centrally managed modern anti-malware software**

Guidance: Azure Lighthouse does not deploy any customer-facing compute resources which could be configured with an anti-malware solution. The underlying infrastructure for the Azure Lighthouse service is handled by Microsoft, which includes managing any installed anti-malware software.

Responsibility: Microsoft

Microsoft Defender for Cloud monitoring: None

## **ES-3: Ensure anti-malware software and signatures are updated**

Guidance: Azure Lighthouse does not deploy any customer-facing compute resources which could be configured with an anti-malware solution. The underlying infrastructure for the Azure Lighthouse service is handled by Microsoft, which includes managing any installed anti-malware software.

Responsibility: Microsoft

Microsoft Defender for Cloud monitoring: None

# Azure Monitor

## Azure Monitor overview

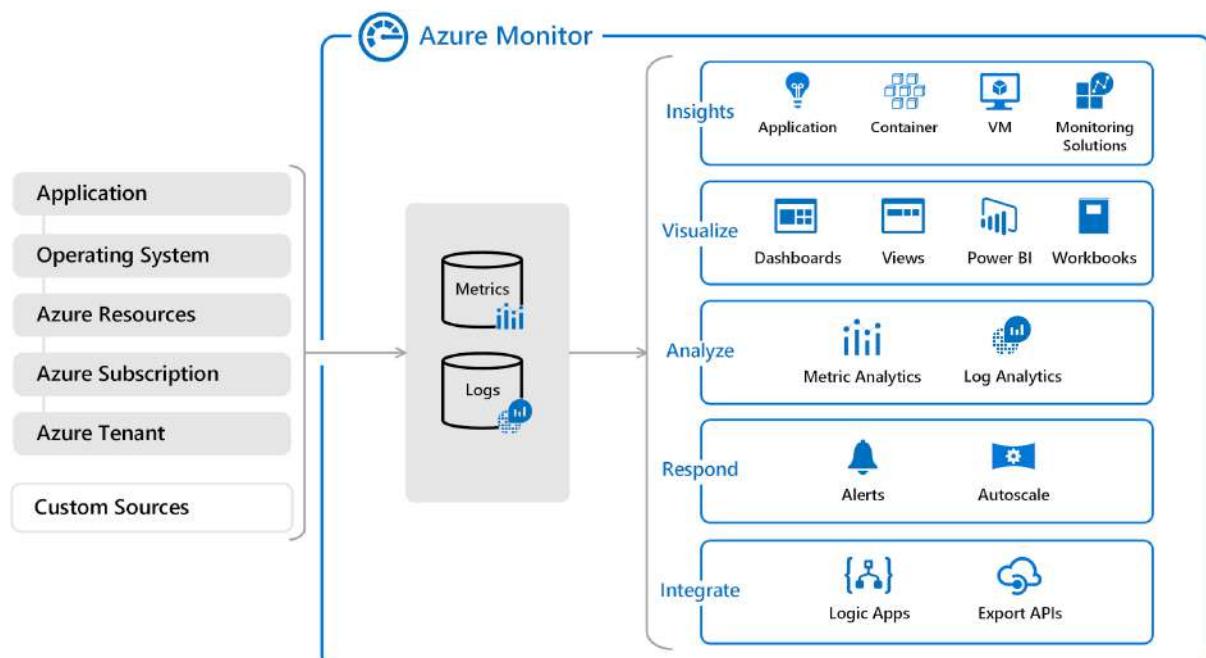
Azure Monitor helps you maximize the availability and performance of your applications and services. It delivers a comprehensive solution for collecting, analyzing, and acting on telemetry from your cloud and on-premises environments. This information helps you understand how your applications are performing and proactively identify issues affecting them and the resources they depend on. Just a few examples of what you can do with Azure Monitor include:

- Detect and diagnose issues across applications and dependencies with Application Insights.
- Correlate infrastructure issues with VM insights and Container insights.
- Drill into your monitoring data with Log Analytics for troubleshooting and deep diagnostics.
- Support operations at scale with smart alerts and automated actions.
- Create visualizations with Azure dashboards and workbooks.
- Collect data from monitored resources using Azure Monitor Metrics.

**Note:** This service supports Azure Lighthouse, which lets service providers sign in to their own tenant to manage subscriptions and resource groups that customers have delegated.

## Overview

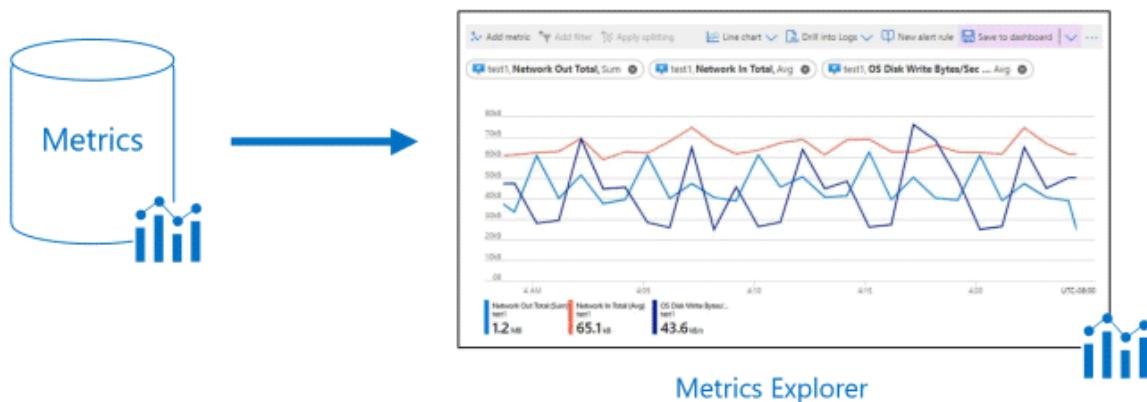
The following diagram gives a high-level view of Azure Monitor. At the center of the diagram are the data stores for metrics and logs, which are the two fundamental types of data used by Azure Monitor. On the left are the sources of monitoring data that populate these data stores. On the right are the different functions that Azure Monitor performs with this collected data.



# Monitoring data platform

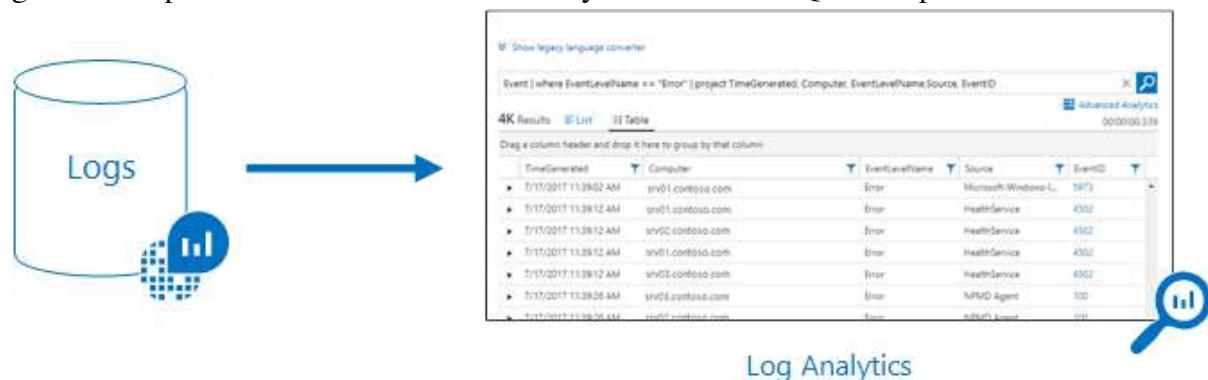
All data collected by Azure Monitor fits into one of two fundamental types, metrics and logs. Metrics are numerical values that describe some aspect of a system at a particular point in time. They are lightweight and capable of supporting near real-time scenarios. Logs contain different kinds of data organized into records with different sets of properties for each type. Telemetry such as events and traces are stored as logs in addition to performance data so that it can all be combined for analysis.

For many Azure resources, you'll see data collected by Azure Monitor right in their Overview page in the Azure portal. Have a look at any virtual machine for example, and you'll see several charts displaying performance metrics. Click on any of the graphs to open the data in metrics explorer in the Azure portal, which allows you to chart the values of multiple metrics over time. You can view the charts interactively or pin them to a dashboard to view them with other visualizations.



Log data collected by Azure Monitor can be analyzed with queries to quickly retrieve, consolidate, and analyze collected data. You can create and test queries using Log Analytics in the Azure portal. You can then either directly analyze the data using different tools or save queries for use with visualizations or alert rules.

Azure Monitor uses a version of the Kusto query language that is suitable for simple log queries but also includes advanced functionality such as aggregations, joins, and smart analytics. You can quickly learn the query language using multiple lessons. Particular guidance is provided to users who are already familiar with SQL and Splunk.



## What data does Azure Monitor collect?

Azure Monitor can collect data from a variety of sources. This ranges from your application, any operating system and services it relies on, down to the platform itself. Azure Monitor collects data from each of the following tiers:

- Application monitoring data: Data about the performance and functionality of the code you have written, regardless of its platform.
- Guest OS monitoring data: Data about the operating system on which your application is running. This could be running in Azure, another cloud, or on-premises.
- Azure resource monitoring data: Data about the operation of an Azure resource. For a complete list of the resources that have metrics or logs.
- Azure subscription monitoring data: Data about the operation and management of an Azure subscription, as well as data about the health and operation of Azure itself.
- Azure tenant monitoring data: Data about the operation of tenant-level Azure services, such as Azure Active Directory.

As soon as you create an Azure subscription and start adding resources such as virtual machines and web apps, Azure Monitor starts collecting data. Activity logs record when resources are created or modified. Metrics tell you how the resource is performing and the resources that it's consuming.

Enable diagnostics to extend the data you're collecting into the internal operation of the resources. Add an agent to compute resources to collect telemetry from their guest operating systems.

Enable monitoring for your application with Application Insights to collect detailed information including page views, application requests, and exceptions. Further verify the availability of your application by configuring an availability test to simulate user traffic.

## Custom sources

Azure Monitor can collect log data from any REST client using the Data Collector API. This allows you to create custom monitoring scenarios and extend monitoring to resources that don't expose telemetry through other sources.

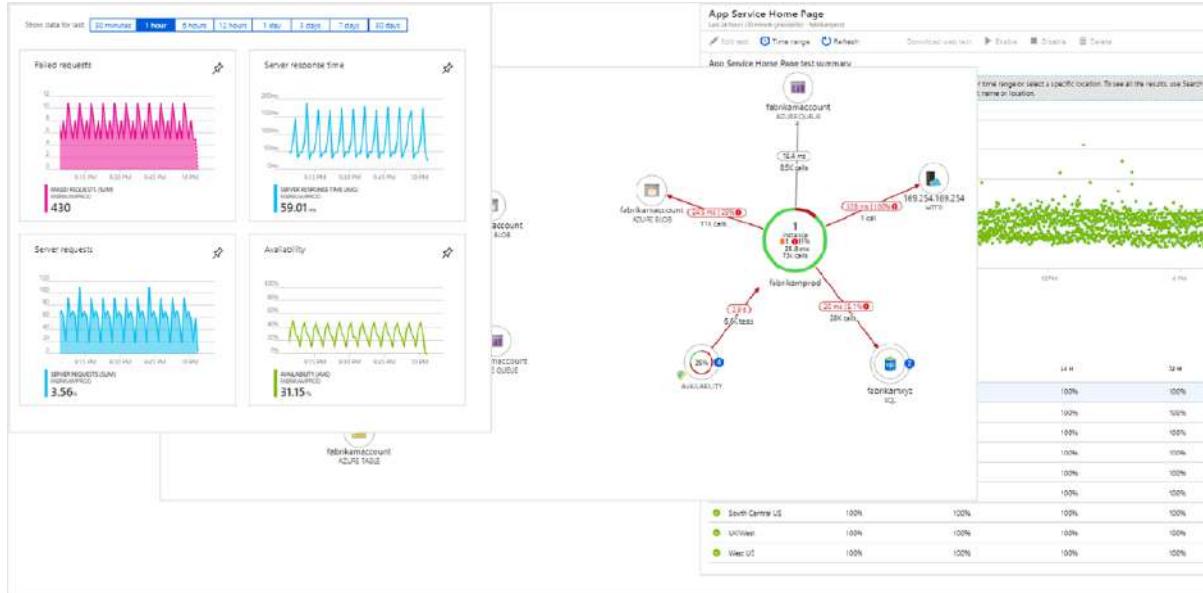
## Insights and curated visualizations

Monitoring data is only useful if it can increase your visibility into the operation of your computing environment. Some Azure resource providers have a "curated visualization" which gives you a customized monitoring experience for that particular service or set of services. They generally require minimal configuration. Larger scalable curated visualizations are known as "insights" and marked with that name in the documentation and Azure portal.

## Application Insights

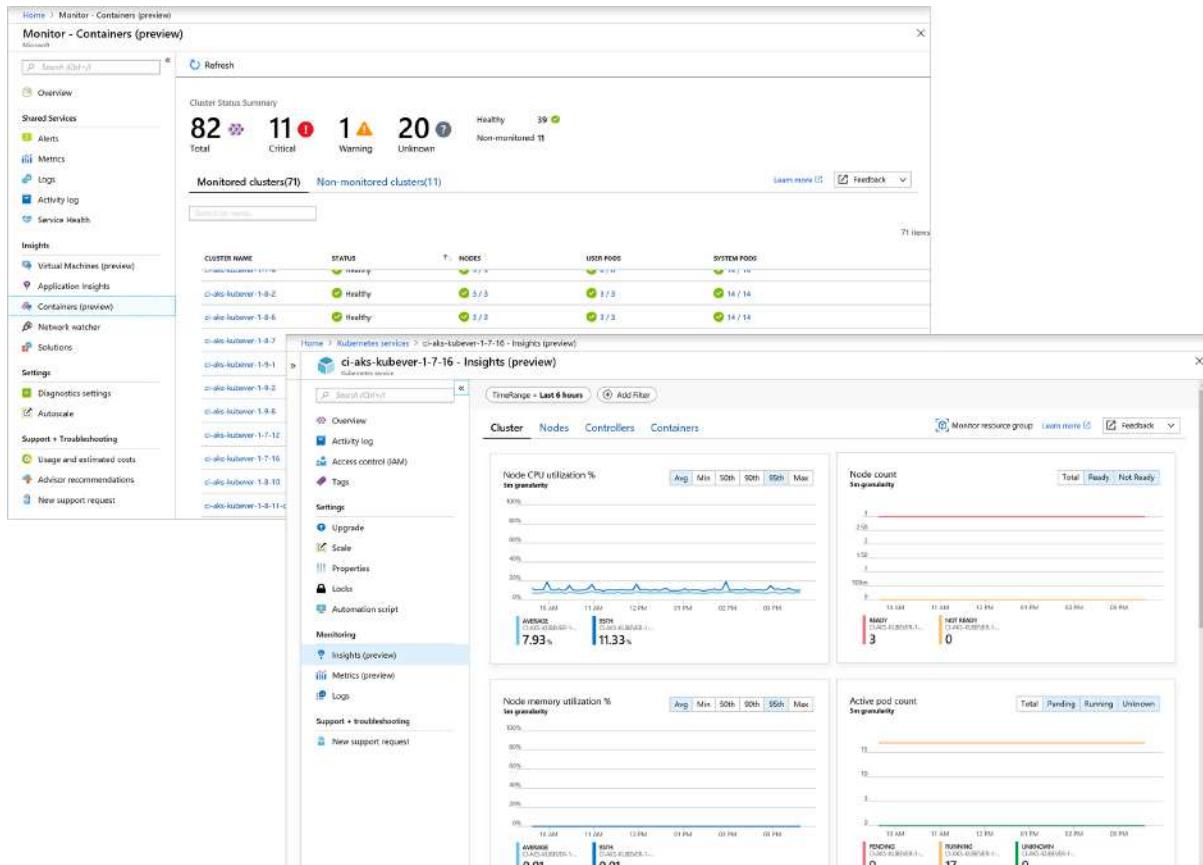
Application Insights monitors the availability, performance, and usage of your web applications whether they're hosted in the cloud or on-premises. It leverages the powerful

data analysis platform in Azure Monitor. It enables you to diagnose errors without waiting for a user to report them. Application Insights includes connection points to a variety of development tools and integrates with Visual Studio to support your DevOps processes.



## Container insights

Container insights monitors the performance of container workloads that are deployed to managed Kubernetes clusters hosted on Azure Kubernetes Service (AKS). After you enable monitoring from Kubernetes clusters, these metrics and logs are automatically collected for you through a containerized version of the Log Analytics agent for Linux.



# VM insights

VM insights monitors your Azure virtual machines (VM) at scale. It analyzes the performance and health of your Windows and Linux VMs and identifies their different processes and interconnected dependencies on external processes. The solution includes support for monitoring performance and application dependencies for VMs hosted on-premises or another cloud provider.

Home > Monitor - Virtual Machines (preview)

## Monitor - Virtual Machines (preview)

Overview

Cloud Services

Alerts

Metric

Metrics (preview)

Logs

Activity log

Service Health

Health

VIRTUAL MACHINES (preview)

- Application insights
- Containers (preview)
- Network watcher
- Storage

Settings

- Diagnostics settings
- Autoscale

SUPPORT + TROUBLESHOOTING

- Usage and estimated costs
- Advisor recommendations
- New support request

### Health

Subscription: Microsoft Azure Region: West Europe Last updated: 1/12/2018 10:23:30 AM

Feedback

Guest VM health

VM distribution by operating system

OPERATING SYSTEM CRITICAL

- Microsoft Windows Server 2016 Datacenter - Unknown
- Red Hat Enterprise Linux Server v7.2 - Unknown

View All

VM distribution by component health

COMPONENT	Critical
CPU	0
Disk	1
Memory	1
Network	0

VM distribution by core services

SERVICE	Critical
DHCP Client	2
DNS Client	2
Print	0

DC01 - Insights (preview)

Last updated: 1/12/2018 10:15:03 AM

Properties

Lads

Automation script

Operations

- Auto-shutdown
- Backup
- Disaster recovery
- Update management
- Inventory
- Change tracking
- State configuration (Preview)
- Run command

Monitoring

- Logs
- Metric
- Diagnostics settings
- Advisor recommendations

Health

Performance

Map

### Platform health

CATEGORY	HEALTH STATUS
Resource health	Available

### Guest VM health

View health diagnosis

VM	HEALTH STATUS
Guest VM health	Unknown

### Component health

COMPONENT	HEALTH STATUS
CPU	Healthy
Disk	Warning
Memory	Critical
Network	Healthy

### Core services health

SERVICE	HEALTH STATUS
DHCP Client	Unknown
DNS Client	Unknown
Print	Critical
RPC Service Health	Critical
Windows Remote Management	Unknown

# **Responding to critical situations**

In addition to allowing you to interactively analyze monitoring data, an effective monitoring solution must be able to proactively respond to critical conditions identified in the data that it collects. This could be sending a text or mail to an administrator responsible for investigating an issue. Or you could launch an automated process that attempts to correct an error condition.

## Alerts

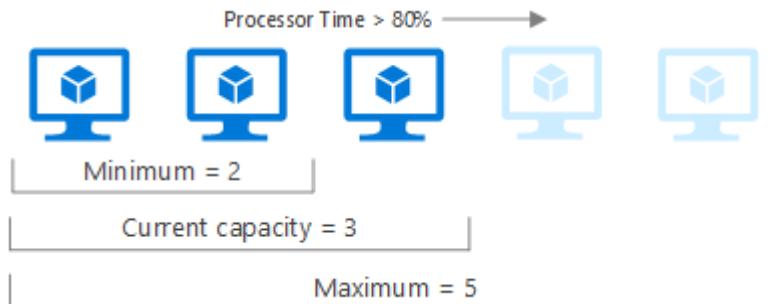
Alerts in Azure Monitor proactively notify you of critical conditions and potentially attempt to take corrective action. Alert rules based on metrics provide near real time alerts based on numeric values. Rules based on logs allow for complex logic across data from multiple sources.

Alert rules in Azure Monitor use action groups, which contain unique sets of recipients and actions that can be shared across multiple rules. Based on your requirements, action groups can perform such actions as using webhooks to have alerts start external actions or to integrate with your ITSM tools.

The screenshot shows the Azure Monitor Alerts dashboard. At the top, it displays subscription information (Contoso IT - demo), resource group (mms-eus), and time range (Past Hour). Below this, key metrics are shown: Total Alerts (29), Smart Groups (1), and Total Alert Rules (15). A note indicates a 96.55% Reduction since August 1, 2018, at 4:38:39 PM. A link to 'Learn More About Alerts' is available. A table below shows the distribution of alerts by severity (Sev 0 to Sev 4) with counts of 26, 0, 3, 0, and 0 respectively. The last row shows 'Enabled 13'. A chart on the right shows a 96.55% reduction in alerts.

## Autoscale

Autoscale allows you to have the right amount of resources running to handle the load on your application. Create rules that use metrics collected by Azure Monitor to determine when to automatically add resources when load increases. Save money by removing resources that are sitting idle. You specify a minimum and maximum number of instances and the logic for when to increase or decrease resources.

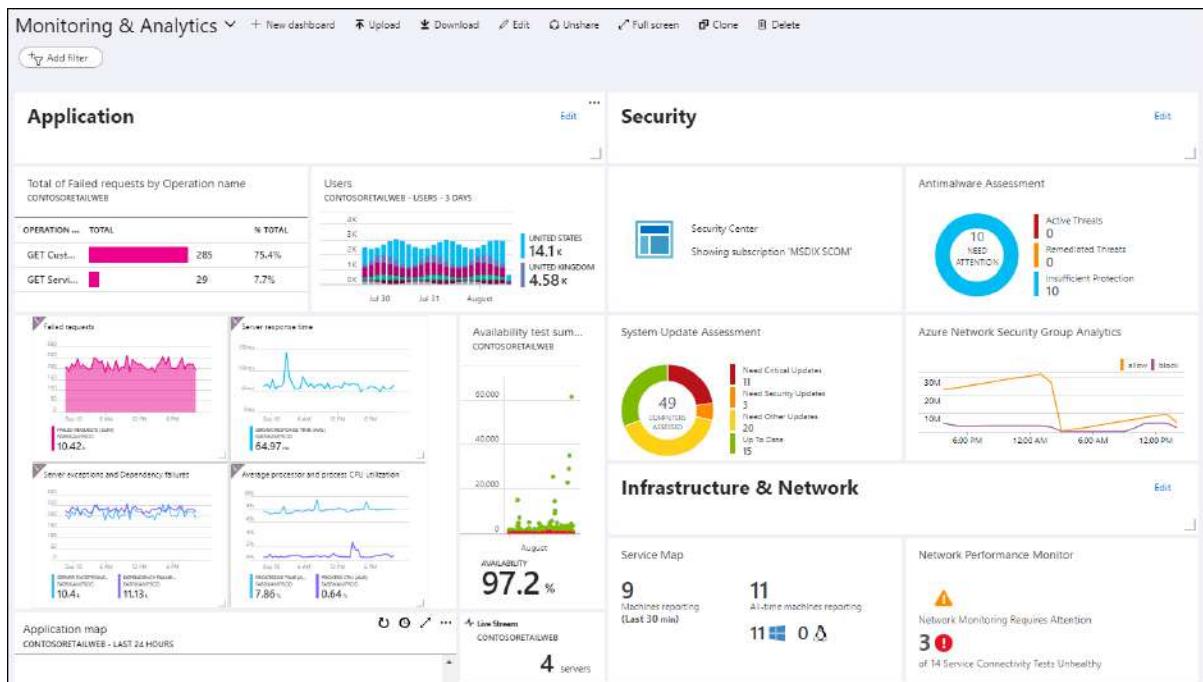


## Visualizing monitoring data

Visualizations such as charts and tables are effective tools for summarizing monitoring data and presenting it to different audiences. Azure Monitor has its own features for visualizing monitoring data and leverages other Azure services for publishing it to different audiences.

## Dashboards

Azure dashboards allow you to combine different kinds of data into a single pane in the Azure portal. You can optionally share the dashboard with other Azure users. Add the output of any log query or metrics chart to an Azure dashboard. For example, you could create a dashboard that combines tiles that show a graph of metrics, a table of activity logs, a usage chart from Application Insights, and the output of a log query.



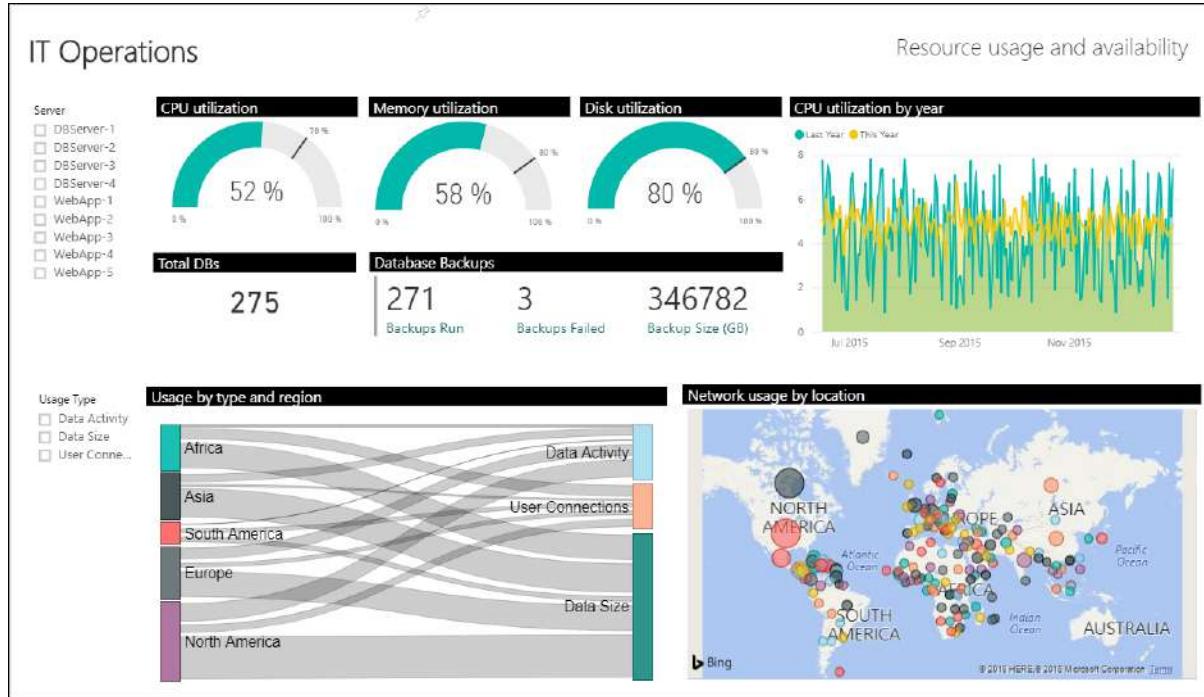
## Workbooks

Workbooks provide a flexible canvas for data analysis and the creation of rich visual reports in the Azure portal. They allow you to tap into multiple data sources from across Azure, and combine them into unified interactive experiences. Use workbooks provided with Insights or create your own from predefined templates.



## Power BI

Power BI is a business analytics service that provides interactive visualizations across a variety of data sources. It's an effective means of making data available to others within and outside your organization. You can configure Power BI to automatically import log data from Azure Monitor to take advantage of these additional visualizations.



## Integrate and export data

You'll often have the requirement to integrate Azure Monitor with other systems and to build custom solutions that use your monitoring data. Other Azure services work with Azure Monitor to provide this integration.

## Event Hub

Azure Event Hubs is a streaming platform and event ingestion service. It can transform and store data using any real-time analytics provider or batching/storage adapters. Use Event Hubs to stream Azure Monitor data to partner SIEM and monitoring tools.

## Logic Apps

Logic Apps is a service that allows you to automate tasks and business processes using workflows that integrate with different systems and services. Activities are available that read and write metrics and logs in Azure Monitor. This allows you to build workflows integrating with a variety of other systems.

## API

Multiple APIs are available to read and write metrics and logs to and from Azure Monitor in addition to accessing generated alerts. You can also configure and retrieve alerts. This provides you with essentially unlimited possibilities to build custom solutions that integrate with Azure Monitor.

## What is monitored by Azure Monitor?

### Insights and curated visualizations

Some services have a curated monitoring experience. That is, Microsoft provides customized functionality meant to act as a starting point for monitoring those services. These experiences are collectively known as curated visualizations with the larger, more complex of them being called Insights.

The experiences collect and analyze a subset of logs and metrics and depending on the service and might also provide out-of-the-box alerting. They present this telemetry in a visual layout. The visualizations vary in size and scale. Some are considered part of Azure Monitor and follow the support and service level agreements for Azure. They are supported in all Azure regions where Azure Monitor is available. Other curated visualizations provide less functionality, might not scale, and might have different agreements. Some might be based solely on Azure Monitor Workbooks, while others might have an extensive custom experience.

The table below lists the available curated visualizations and more detailed information about them.

**Note:** Another type of older visualization called monitoring solutions are no longer in active development. The replacement technology is the Azure Monitor Insights mentioned above. We suggest you use the insights and not deploy new instances of solutions.

Name with docs link	State	Description
Azure Monitor Workbooks for Azure Active Directory	GA (General availability )	Azure Active Directory provides workbooks to understand the effect of your Conditional Access policies, to troubleshoot sign-in failures, and to identify legacy authentications.

Azure Backup Insights	GA	Provides built-in monitoring and alerting capabilities in a Recovery Services vault.
Azure Monitor for Azure Cache for Redis (preview)	GA	Provides a unified, interactive view of overall performance, failures, capacity, and operational health
Azure Cosmos DB Insights	GA	Provides a view of the overall performance, failures, capacity, and operational health of all your Azure Cosmos DB resources in a unified interactive experience.
Azure Data Explorer insights	GA	Azure Data Explorer Insights provides comprehensive monitoring of your clusters by delivering a unified view of your cluster performance, operations, usage, and failures.
Azure HDInsight (preview)	Preview	An Azure Monitor workbook that collects important performance metrics from your HDInsight cluster and provides the visualizations and dashboards for most common scenarios. Gives a complete view of a single HDInsight cluster including resource utilization and application status
Azure IoT Edge Insights	GA	Visualize and explore metrics collected from the IoT Edge device right in the Azure portal using Azure Monitor Workbooks based public templates. The curated workbooks use built-in metrics from the IoT Edge runtime. These views don't need any metrics instrumentation from the workload modules.
Azure Key Vault Insights (preview)	GA	Provides comprehensive monitoring of your key vaults by delivering a unified view of your Key Vault requests, performance, failures, and latency.

Azure Monitor Application Insights	GA	Extensible Application Performance Management (APM) service which monitors the availability, performance, and usage of your web applications whether they're hosted in the cloud or on-premises. It leverages the powerful data analysis platform in Azure Monitor to provide you with deep insights into your application's operations. It enables you to diagnose errors without waiting for a user to report them. Application Insights includes connection points to a variety of development tools and integrates with Visual Studio to support your DevOps processes.
Azure Monitor Log Analytics Workspace	Preview	Log Analytics Workspace Insights (preview) provides comprehensive monitoring of your workspaces through a unified view of your workspace usage, performance, health, agent, queries, and change log. This article will help you understand how to onboard and use Log Analytics Workspace Insights (preview).
Azure Service Bus	Preview	
Azure SQL insights	GA	A comprehensive interface for monitoring any product in the Azure SQL family. SQL insights uses dynamic management views to expose the data you need to monitor health, diagnose problems, and tune performance. Note: If you are just setting up SQL monitoring, use this instead of the SQL Analytics solution.
Azure Storage Insights	GA	Provides comprehensive monitoring of your Azure Storage accounts by delivering a unified view of your Azure Storage services performance, capacity, and availability.
Azure VM Insights	GA	Monitors your Azure virtual machines (VM) and virtual machine scale sets at scale. It analyzes the performance and health of your Windows and Linux VMs, and monitors their processes and dependencies on other resources and external processes.

---

Azure Network Insights	GA	Provides a comprehensive view of health and metrics for all your network resource. The advanced search capability helps you identify resource dependencies, enabling scenarios like identifying resource that are hosting your website, by simply searching for your website name.
Azure Container Insights	GA	Monitors the performance of container workloads that are deployed to managed Kubernetes clusters hosted on Azure Kubernetes Service (AKS). It gives you performance visibility by collecting metrics from controllers, nodes, and containers that are available in Kubernetes through the Metrics API. Container logs are also collected. After you enable monitoring from Kubernetes clusters, these metrics and logs are automatically collected for you through a containerized version of the Log Analytics agent for Linux.
Azure Monitor for Resource Groups	GA	Triage and diagnose any problems your individual resources encounter, while offering context as to the health and performance of the resource group as a whole.
Azure Monitor SAP	GA	An Azure-native monitoring product for anyone running their SAP landscapes on Azure. It works with both SAP on Azure Virtual Machines and SAP on Azure Large Instances. Collects telemetry data from Azure infrastructure and databases in one central location and visually correlate the data for faster troubleshooting. You can monitor different components of an SAP landscape, such as Azure virtual machines (VMs), high-availability cluster, SAP HANA database, SAP NetWeaver, and so on, by adding the corresponding provider for that component.

---

---

Azure Stack HCI insights	Preview	Azure Monitor Workbook based. Provides health, performance, and usage insights about registered Azure Stack HCI, version 21H2 clusters that are connected to Azure and are enrolled in monitoring. It stores its data in a Log Analytics workspace, which allows it to deliver powerful aggregation and filtering and analyze data trends over time.
Windows Virtual Desktop Insights	GA	Azure Monitor for Windows Virtual Desktop (preview) is a dashboard built on Azure Monitor Workbooks that helps IT professionals understand their Windows Virtual Desktop environments. This topic will walk you through how to set up Azure Monitor for Windows Virtual Desktop to monitor your Windows Virtual Desktop environments.

## Product integrations

The other services and older monitoring solutions in the following table store their data in a Log Analytics workspace so that it can be analyzed with other log data collected by Azure Monitor.

Product/Service	Description
Azure Automation	Manage operating system updates and track changes on Windows and Linux computers.
Azure Information Protection	Classify and optionally protect documents and emails.
Azure Security Center	Collect and analyze security events and perform threat analysis.

---

Azure Sentinel	Connects to different sources including Office 365 and Amazon Web Services Cloud Trail.
Microsoft Intune	Create a diagnostic setting to send logs to Azure Monitor.
Network Traffic Analytics	Analyzes Network Watcher network security group (NSG) flow logs to provide insights into traffic flow in your Azure cloud.
System Center Operations Manager	Collect data from Operations Manager agents by connecting their management group to Azure Monitor. Assess the risk and health of your System Center Operations Manager management group with Operations Manager Assessment solution.
Microsoft Teams Rooms	Integrated, end-to-end management of Microsoft Teams Rooms devices.
Visual Studio App Center	Build, test, and distribute applications and then monitor their status and usage.

---

---

Windows	Windows Update Compliance - Assess your Windows desktop upgrades. Desktop Analytics - Integrates with Configuration Manager to provide insight and intelligence to make more informed decisions about the update readiness of your Windows clients.
---------	--

---

The following solutions also integrate with parts of Azure Monitor. Note that solutions, are no longer under active development. Use insights instead.

---

Network - Network Performance Monitor solution

---

Network - Azure Application Gateway Solution

---

Office 365 solution

Monitor your Office 365 environment. Updated version with improved onboarding available through Microsoft Sentinel.

---

SQL Analytics solution

Use SQL Insights instead

---

Surface Hub solution

## Third-party integration

Integration	Description
-------------	-------------

---

---

ITSM	The IT Service Management Connector (ITSMC) allows you to connect Azure and a supported IT Service Management (ITSM) product/service.
Azure Monitor Partners	A list of partners that integrate with Azure Monitor in some form
Azure Monitor Partner integrations	Specialized integrations between Azure Monitor and other non-Microsoft monitoring platforms if you've already built on them. Examples include Datadog and Elastic

## Resources outside of Azure

Azure Monitor can collect data from resources outside of Azure using the methods listed in the following table.

Resource	Method
Application s	Monitor web applications outside of Azure using Application Insights.
Virtual machines	Use agents to collect data from the guest operating system of virtual machines in other cloud environments or on-premises.
REST API Client	Separate APIs are available to write data to Azure Monitor Logs and Metrics from any REST API client.

## Azure supported services

The following table lists Azure services and the data they collect into Azure Monitor.

- Metrics - The service automatically collects metrics into Azure Monitor Metrics.

- Logs - The service supports diagnostic settings which can send metrics and platform logs into Azure Monitor Logs for analysis in Log Analytics.
- Insight - There is an insight available which provides a customized monitoring experience for the service.

Service	Resource Provider Namespace	Has Metrics	Has Logs
Azure Active Directory Domain Services	Microsoft.AAD/DomainServices	No	Yes
Azure Active Directory	Microsoft.Aadiam/azureADMetrics	Yes	No
Azure Analysis Services	Microsoft.AnalysisServices/servers	Yes	Yes
API Management	Microsoft.ApiManagement/service	Yes	Yes
Azure App Configuration	Microsoft.AppConfiguration/configurationStores	Yes	Yes
Azure Spring Cloud	Microsoft.AppPlatform/Spring	Yes	Yes

---

Azure Attestation Service	Microsoft.Attestation/attestationProviders	No	Yes
---------------------------	--	----	-----

---

Azure Automation	Microsoft.Automation/automationAccounts	Yes	Yes
------------------	---	-----	-----

---

Azure VMware Solution	Microsoft.AVS/privateClouds	Yes	Yes
-----------------------	-----------------------------	-----	-----

---

Azure Batch	Microsoft.Batch/batchAccounts	Yes	Yes
-------------	-------------------------------	-----	-----

---

Azure Batch	Microsoft.BatchAI/workspaces	No	No
-------------	------------------------------	----	----

---

Azure Cognitive Services-Bing Search API	Microsoft.Bing/accounts	Yes	No
--	-------------------------	-----	----

---

Azure Blockchain Service	Microsoft.Blockchain/blockchainMembers	Yes	Yes
--------------------------	--	-----	-----

---

Azure Blockchain Service	Microsoft.Blockchain/cordaMembers	No	Yes
--------------------------	-----------------------------------	----	-----

---

Azure Bot Service	Microsoft.BotService/botServices	Yes	Yes
-------------------	----------------------------------	-----	-----

---

---

Azure Cache for Redis	Microsoft.Cache/Redis	Yes	Yes
-----------------------	-----------------------	-----	-----

---

Azure Cache for Redis	Microsoft.Cache/redisEnterprise	Yes	No
-----------------------	---------------------------------	-----	----

---

Content Delivery Network	Microsoft.Cdn/CdnWebApplicationFirewallPolicies	Yes	Yes
--------------------------	---	-----	-----

---

Content Delivery Network	Microsoft.Cdn/profiles	Yes	Yes
--------------------------	------------------------	-----	-----

---

Content Delivery Network	Microsoft.Cdn/profiles/endpoints	No	Yes
--------------------------	----------------------------------	----	-----

---

Azure Virtual Machines - Classic	Microsoft.ClassicCompute/domainNames/slots/roles	Yes	No
----------------------------------	--	-----	----

---

---

Azure Virtual Machines - Classic	Microsoft.ClassicCompute/ virtualMachines	Yes	No
---	--	-----	----

---

Virtual Network (Classic)	Microsoft.ClassicNetwork/ networkSecurityGroups	No	Yes
---------------------------------	--	----	-----

---

Azure Storage (Classic)	Microsoft.ClassicStorage/ storageAccounts	Yes	No
-------------------------------	--	-----	----

---

Azure Storage Blobs (Classic)	Microsoft.ClassicStorage/ storageAccounts/blobServices	Yes	No
--	---	-----	----

---

Azure Storage Files (Classic)	Microsoft.ClassicStorage/ storageAccounts/fileServices	Yes	No
--	---	-----	----

---

Azure Storage Queues (Classic)	Microsoft.ClassicStorage/ storageAccounts/queueServices	Yes	No
---	--	-----	----

---

Azure Storage Tables (Classic)	Microsoft.ClassicStorage/ storageAccounts/tableServices	Yes	No
---	--	-----	----

---

Microsoft Cloud Test Platform	Microsoft.Cloudtest/hostedpools	Yes	No
-------------------------------------	---------------------------------	-----	----

---

Microsoft Cloud Test Platform	Microsoft.Cloudtest/pools	Yes	No
-------------------------------------	---------------------------	-----	----

---

---

Cray ClusterStorage in Azure	Microsoft.ClusterStor/nodes	Yes	No
------------------------------	-----------------------------	-----	----

---

Azure Cognitive Services	Microsoft.CognitiveServices/accounts	Yes	Yes
--------------------------	--------------------------------------	-----	-----

---

Azure Communication Services	Microsoft.Communication/CommunicationServices	Yes	Yes
------------------------------	---	-----	-----

---

Azure Cloud Services	Microsoft.Compute/cloudServices	Yes	No
----------------------	---------------------------------	-----	----

---

Azure Cloud Services	Microsoft.Compute/cloudServices/roles	Yes	No
----------------------	---------------------------------------	-----	----

---

---

Azure Virtual Machines Virtual Machine Scale Sets	Microsoft.Compute/disks	Yes	No
--	-------------------------	-----	----

---

Azure Virtual Machines Virtual Machine Scale Sets	Microsoft.Compute/virtualMachines	Yes	No
--	-----------------------------------	-----	----

---

Azure Virtual Machines Virtual Machine Scale Sets	Microsoft.Compute/ virtualMachineScaleSets	Yes	No
--	---	-----	----

---

Azure Virtual Machines Virtual Machine Scale Sets	Microsoft.Compute/ virtualMachineScaleSets/ virtualMachines	Yes	No
--	---	-----	----

---

---

Microsoft Connected Vehicle Platform	Microsoft.ConnectedVehicle/platformAccounts	Yes	Yes
--------------------------------------	---	-----	-----

---

Azure Container Instances	Microsoft.ContainerInstance/containerGroups	Yes	No
---------------------------	---	-----	----

---

Azure Container Registry	Microsoft.ContainerRegistry/registries	Yes	Yes
--------------------------	--	-----	-----

---

Azure Kubernetes Service (AKS)	Microsoft.ContainerService/managedClusters	Yes	Yes
--------------------------------	--	-----	-----

---

Azure Custom Providers	Microsoft.CustomProviders/resourceProviders	Yes	Yes
------------------------	---	-----	-----

---

Microsoft Dynamics 365 Customer Insights	Microsoft.D365CustomerInsights/instances	No	Yes
--	--	----	-----

---

Azure Stack Edge	Microsoft.DataBoxEdge/DataBoxEdgeDevices	Yes	No
------------------	--	-----	----

---

Azure Databricks	Microsoft.Databricks/workspaces	No	Yes
------------------	---------------------------------	----	-----

---

--	--	--	--

---

Project CI	Microsoft.DataCollaboration/ workspaces	Yes	Yes
------------	--	-----	-----

---

Azure Data Factory	Microsoft.DataFactory/dataFactories	Yes	No
--------------------------	-------------------------------------	-----	----

---

Azure Data Factory	Microsoft.DataFactory/factories	Yes	Yes
--------------------------	---------------------------------	-----	-----

---

Azure Data Lake Analytics	Microsoft.DataLakeAnalytics/ accounts	Yes	Yes
---------------------------------	--	-----	-----

---

Azure Data Lake Storage Gen2	Microsoft.DataLakeStore/accounts	Yes	Yes
---------------------------------------	----------------------------------	-----	-----

---

Azure Data Share	Microsoft.DataShare/accounts	Yes	Yes
------------------------	------------------------------	-----	-----

---

Azure Database for MariaDB	Microsoft.DBforMariaDB/servers	Yes	Yes
-------------------------------------	--------------------------------	-----	-----

---

Azure Database for MySQL	Microsoft.DBforMySQL/ flexibleServers	Yes	Yes
--------------------------------	--	-----	-----

---

Azure Database for MySQL	Microsoft.DBforMySQL/servers	Yes	Yes
--------------------------------	------------------------------	-----	-----

---

Azure Database for PostgreSQL	Microsoft.DBforPostgreSQL/flexibleServers	Yes	Yes
Azure Database for PostgreSQL	Microsoft.DBforPostgreSQL/serverGroupsV2	Yes	Yes
Azure Database for PostgreSQL	Microsoft.DBforPostgreSQL/servers	Yes	Yes
Azure Database for PostgreSQL	Microsoft.DBforPostgreSQL/serversV2	Yes	Yes
Microsoft Windows Virtual Desktop	Microsoft.DesktopVirtualization/applicationgroups	No	Yes
Microsoft Windows Virtual Desktop	Microsoft.DesktopVirtualization/hostpools	No	Yes
Microsoft Windows Virtual Desktop	Microsoft.DesktopVirtualization/workspaces	No	Yes
Azure IoT Hub	Microsoft.Devices/ElasticPools	Yes	No

---

Azure IoT Hub	Microsoft.Devices/ElasticPools/iotHubTenants	Yes	Yes
---------------	--	-----	-----

---

Azure IoT Hub	Microsoft.Devices/IotHubs	Yes	Yes
---------------	---------------------------	-----	-----

---

Azure IoT Hub Device Provisioning Service	Microsoft.Devices/ ProvisioningServices	Yes	Yes
--	--	-----	-----

---

Azure Digital Twins	Microsoft.DigitalTwins/ digitalTwinsInstances	Yes	Yes
---------------------	--	-----	-----

---

Azure Cosmos DB	Microsoft.DocumentDB/ databaseAccounts	Yes	Yes
-----------------	---	-----	-----

---

Azure Grid	Microsoft.EventGrid/domains	Yes	Yes
------------	-----------------------------	-----	-----

---

Azure Grid	Microsoft.EventGrid/ eventSubscriptions	Yes	No
------------	--	-----	----

---

Azure Grid	Microsoft.EventGrid/extensionTopics	Yes	No
------------	-------------------------------------	-----	----

---

Azure Grid	Microsoft.EventGrid/ partnerNamespaces	Yes	Yes
------------	---	-----	-----

---

Azure Grid	Microsoft.EventGrid/partnerTopics	Yes	Yes
------------	-----------------------------------	-----	-----

---

---

Azure Grid	Microsoft.EventGrid/systemTopics	Yes	Yes
------------	----------------------------------	-----	-----

---

Azure Grid	Microsoft.EventGrid/topics	Yes	Yes
------------	----------------------------	-----	-----

---

Azure Event Hubs	Microsoft.EventHub/clusters	Yes	No
------------------	-----------------------------	-----	----

---

Azure Event Hubs	Microsoft.EventHub/namespaces	Yes	Yes
------------------	-------------------------------	-----	-----

---

Microsoft Experimentation Platform	microsoft.experimentation/experimentWorkspaces	Yes	Yes
------------------------------------	--	-----	-----

---

Azure HDInsight	Microsoft.HDInsight/clusters	Yes	No
-----------------	------------------------------	-----	----

---

Azure API for FHIR	Microsoft.HealthcareApis/services	Yes	Yes
--------------------	-----------------------------------	-----	-----

---

Azure API for FHIR	Microsoft.HealthcareApis/workspaces/iotconnectors	Yes	No
--------------------	---	-----	----

---

StorSimple	microsoft.hybridnetwork/networkfunctions	Yes	No
------------	--	-----	----

---

---

StorSimpl e	microsoft.hybridnetwork/virtualnetworkfunctions	Yes	No
-------------	---	-----	----

---

Azure Monitor	microsoft.insights/autoscalesettings	Yes	Yes
---------------	--------------------------------------	-----	-----

---

Azure Monitor	microsoft.insights/components	Yes	Yes
---------------	-------------------------------	-----	-----

---

Azure IoT Central	Microsoft.IoTCentral/IoTApps	Yes	No
-------------------	------------------------------	-----	----

---

Azure Key Vault	Microsoft.KeyVault/managedHSMs	Yes	Yes
-----------------	--------------------------------	-----	-----

---

Azure Key Vault	Microsoft.KeyVault/vaults	Yes	Yes
-----------------	---------------------------	-----	-----

---

Azure Kubernetes Service (AKS)	Microsoft.Kubernetes/connectedClusters	Yes	No
--------------------------------	--	-----	----

---

Azure Data Explorer	Microsoft.Kusto/clusters	Yes	Yes
---------------------	--------------------------	-----	-----

---

Azure Logic Apps	Microsoft.Logic/integrationAccounts	No	Yes
Azure Logic Apps	Microsoft.Logic/integrationServiceEnvironments	Yes	No
Azure Logic Apps	Microsoft.Logic/workflows	Yes	Yes
Azure Machine Learning	Microsoft.MachineLearningServices/ workspaces	Yes	Yes
Azure Maps	MicrosoftMaps/accounts	Yes	No
Azure Media Services	Microsoft.Media/mediaservices	Yes	Yes
Azure Media Services	Microsoft.Media/mediaservices/ liveEvents	Yes	No
Azure Media Services	Microsoft.Media/mediaservices/ streamingEndpoints	Yes	No
Azure Media Services	Microsoft.Media/videoAnalyzers	Yes	Yes

Azure Spatial Anchors	Microsoft.MixedReality/ remoteRenderingAccounts	Yes	No
Azure Spatial Anchors	Microsoft.MixedReality/ spatialAnchorsAccounts	Yes	No
Azure NetApp Files	Microsoft.NetApp/netAppAccounts/ capacityPools	Yes	No
Azure NetApp Files	Microsoft.NetApp/netAppAccounts/ capacityPools/volumes	Yes	No
Application Gateway	Microsoft.Network/ applicationGateways	Yes	Yes
Azure Firewall	Microsoft.Network/azureFirewalls	Yes	Yes
Azure Bastion	Microsoft.Network/bastionHosts	Yes	Yes
VPN Gateway	Microsoft.Network/connections	Yes	No
Azure DNS	Microsoft.Network/dnszones	Yes	No
Azure ExpressR oute	Microsoft.Network/ expressRouteCircuits	Yes	Yes

---

Azure ExpressRoute	Microsoft.Network/expressRouteGateways	Yes	No
--------------------	--	-----	----

---

Azure ExpressRoute	Microsoft.Network/expressRoutePorts	Yes	No
--------------------	-------------------------------------	-----	----

---

Azure Front Door	Microsoft.Network/frontdoors	Yes	Yes
------------------	------------------------------	-----	-----

---

Azure Load Balancer	Microsoft.Network/loadBalancers	Yes	Yes
---------------------	---------------------------------	-----	-----

---

Azure Load Balancer	Microsoft.Network/natGateways	Yes	No
---------------------	-------------------------------	-----	----

---

Azure Virtual Network	Microsoft.Network/networkInterfaces	Yes	No
-----------------------	-------------------------------------	-----	----

---

Azure Virtual Network	Microsoft.Network/networkSecurityGroups	No	Yes
-----------------------	---	----	-----

---

Azure Network Watcher	Microsoft.Network/networkWatchers/connectionMonitors	Yes	No
-----------------------	--	-----	----

---

Azure Virtual WAN	Microsoft.Network/p2sVpnGateways	Yes	Yes
-------------------	----------------------------------	-----	-----

---

Azure DNS Private Zones	Microsoft.Network/privateDnsZones	Yes	No
Azure Private Link	Microsoft.Network/privateEndpoints	Yes	No
Azure Private Link	Microsoft.Network/ privateLinkServices	Yes	No
Azure Virtual Network	Microsoft.Network/ publicIPAddresses	Yes	Yes
Azure Traffic Manager	Microsoft.Network/ trafficmanagerprofiles	Yes	Yes
Azure Virtual WAN	Microsoft.Network/virtualHubs	Yes	No
Azure VPN Gateway	Microsoft.Network/ virtualNetworkGateways	Yes	Yes
Azure Virtual Network	Microsoft.Network/virtualNetworks	Yes	Yes
Azure Virtual Network	Microsoft.Network/virtualRouters	Yes	No

Azure Virtual WAN	Microsoft.Network/vpnGateways	Yes	Yes
Azure Notification Hubs	Microsoft.NotificationHubs/namespaces/notificationHubs	Yes	No
Azure Monitor	Microsoft.OperationalInsights/workspaces	Yes	Yes
Azure Peering Service	Microsoft.Peering/peerings	Yes	No
Azure Peering Service	Microsoft.Peering/peeringServices	Yes	No
Microsoft Power BI	Microsoft.PowerBI/tenants	No	Yes
Microsoft Power BI	Microsoft.PowerBI/tenants/workspaces	No	Yes
Power BI Embedded	Microsoft.PowerBIDedicated/capacities	Yes	Yes
Azure Purview	Microsoft.Purview/accounts	Yes	Yes

---

Azure Site Recovery	Microsoft.RecoveryServices/vaults	Yes	Yes
---------------------	-----------------------------------	-----	-----

---

Azure Relay	Microsoft.Relay/namespaces	Yes	Yes
-------------	----------------------------	-----	-----

---

Azure Resource Manager	Microsoft.Resources/subscriptions	Yes	No
------------------------	-----------------------------------	-----	----

---

Azure Cognitive Search	Microsoft.Search/searchServices	Yes	Yes
------------------------	---------------------------------	-----	-----

---

Azure Service Bus	Microsoft.ServiceBus/namespaces	Yes	Yes
-------------------	---------------------------------	-----	-----

---

Service Fabric	Microsoft.ServiceFabric	No	No
----------------	-------------------------	----	----

---

Azure SignalR Service	Microsoft.SignalRService/SignalR	Yes	Yes
-----------------------	----------------------------------	-----	-----

---

Azure SignalR Service	Microsoft.SignalRService/WebPubSub	Yes	Yes
-----------------------	------------------------------------	-----	-----

---

Azure SQL Managed Instance	Microsoft.Sql/managedInstances	Yes	Yes
----------------------------	--------------------------------	-----	-----

---

---

Azure SQL Database	Microsoft.Sql/servers/databases	Yes	No
--------------------------	---------------------------------	-----	----

---

Azure SQL Database	Microsoft.Sql/servers/elasticpools	Yes	No
--------------------------	------------------------------------	-----	----

---

Azure Storage	Microsoft.Storage/storageAccounts	Yes	No
------------------	-----------------------------------	-----	----

---

Azure Storage Blobs	Microsoft.Storage/storageAccounts/ blobServices	Yes	Yes
---------------------------	--	-----	-----

---

Azure Storage Files	Microsoft.Storage/storageAccounts/ fileServices	Yes	Yes
---------------------------	--	-----	-----

---

Azure Storage Queue Services	Microsoft.Storage/storageAccounts/ queueServices	Yes	Yes
---------------------------------------	---	-----	-----

---

Azure Table Services	Microsoft.Storage/storageAccounts/ tableServices	Yes	Yes
----------------------------	---	-----	-----

---

Azure HPC Cache	Microsoft.StorageCache/caches	Yes	No
-----------------------	-------------------------------	-----	----

---

---

Azure Storage	Microsoft.StorageSync/storageSyncServices	Yes	No
---------------	---	-----	----

---

Azure Stream Analytics	Microsoft.StreamAnalytics/streamingjobs	Yes	Yes
------------------------	---	-----	-----

---

Azure Synapse Analytics	Microsoft.Synapse/workspaces	Yes	Yes
-------------------------	------------------------------	-----	-----

---

Azure Synapse Analytics	Microsoft.Synapse/workspaces/bigDataPools	Yes	Yes
-------------------------	---	-----	-----

---

Azure Synapse Analytics	Microsoft.Synapse/workspaces/sqlPools	Yes	Yes
-------------------------	---------------------------------------	-----	-----

---

Azure Time Series Insights	Microsoft.TimeSeriesInsights/environments	Yes	Yes
----------------------------	---	-----	-----

---

Azure Time Series Insights	Microsoft.TimeSeriesInsights/environments/eventsources	Yes	Yes
----------------------------	--	-----	-----

---

Azure VMware Solution	Microsoft.VMwareCloudSimple/virtualMachines	Yes	No
-----------------------	---	-----	----

---

Azure App Service Azure Functions	Microsoft.Web/connections	Yes	No
--------------------------------------	---------------------------	-----	----

---

Azure App Service	Microsoft.Web/hostingEnvironments	Yes	Yes
Azure Functions			
Azure App Service	Microsoft.Web/hostingEnvironments/ multiRolePools	Yes	No
Azure Functions			
Azure App Service	Microsoft.Web/hostingEnvironments/ workerPools	Yes	No
Azure Functions			
Azure App Service	Microsoft.Web/serverFarms	Yes	No
Azure Functions			
Azure App Service	Microsoft.Web/sites	Yes	Yes
Azure Functions			
Azure App Service	Microsoft.Web/sites/slots	Yes	Yes
Azure Functions			
Azure App Service	Microsoft.Web/staticSites	Yes	No
Azure Functions			

# Azure Policy

## What is Azure Policy?

Azure Policy helps to enforce organizational standards and to assess compliance at scale. Through its compliance dashboard, it provides an aggregated view to evaluate the overall state of the environment, with the ability to drill down to the per-resource, per-policy granularity. It also helps to bring your resources to compliance through bulk remediation for existing resources and automatic remediation for new resources.

All Azure Policy data and objects are encrypted at rest.

## Overview

Azure Policy evaluates resources in Azure by comparing the properties of those resources to business rules. These business rules, described in JSON format, are known as policy definitions. To simplify management, several business rules can be grouped together to form a policy initiative (sometimes called a policySet). Once your business rules have been formed, the policy definition or initiative is assigned to any scope of resources that Azure supports, such as management groups, subscriptions, resource groups, or individual resources. The assignment applies to all resources within the Resource Manager scope of that assignment. Subscopes can be excluded, if necessary.

Azure Policy uses a JSON format to form the logic the evaluation uses to determine whether a resource is compliant or not. Definitions include metadata and the policy rule. The defined rule can use functions, parameters, logical operators, conditions, and property aliases to match exactly the scenario you want. The policy rule determines which resources in the scope of the assignment get evaluated.

## Understand evaluation outcomes

Resources are evaluated at specific times during the resource lifecycle, the policy assignment lifecycle, and for regular ongoing compliance evaluation. The following are the times or events that cause a resource to be evaluated:

- A resource is created, updated, or deleted in a scope with a policy assignment.
- A policy or initiative is newly assigned to a scope.
- A policy or initiative already assigned to a scope is updated.
- During the standard compliance evaluation cycle, which occurs once every 24 hours.

## Control the response to an evaluation

Business rules for handling non-compliant resources vary widely between organizations. Examples of how an organization wants the platform to respond to a non-compliant resource include:

- Deny the resource change
- Log the change to the resource

- Alter the resource before the change
- Alter the resource after the change
- Deploy related compliant resources

Azure Policy makes each of these business responses possible through the application of effects. Effects are set in the policy rule portion of the policy definition.

## Remediate non-compliant resources

While these effects primarily affect a resource when the resource is created or updated, Azure Policy also supports dealing with existing non-compliant resources without needing to alter that resource.

## Getting started

### Azure Policy and Azure RBAC

There are a few key differences between Azure Policy and Azure role-based access control (Azure RBAC). Azure Policy evaluates state by examining properties on resources that are represented in Resource Manager and properties of some Resource Providers. Azure Policy doesn't restrict actions (also called operations). Azure Policy ensures that resource state is compliant to your business rules without concern for who made the change or who has permission to make a change. Some Azure Policy resources, such as policy definitions, initiative definitions, and assignments, are visible to all users. This design enables transparency to all users and services for what policy rules are set in their environment.

Azure RBAC focuses on managing user actions at different scopes. If control of an action is required, then Azure RBAC is the correct tool to use. Even if an individual has access to perform an action, if the result is a non-compliant resource, Azure Policy still blocks the create or update.

The combination of Azure RBAC and Azure Policy provides full scope control in Azure.

### Azure RBAC permissions in Azure Policy

Azure Policy has several permissions, known as operations, in two Resource Providers:

- Microsoft.Authorization
- Microsoft.PolicyInsights

Many Built-in roles grant permission to Azure Policy resources. The Resource Policy Contributor role includes most Azure Policy operations. Owner has full rights. Both Contributor and Reader have access to all read Azure Policy operations. Contributor may trigger resource remediation, but can't create definitions or assignments. User Access Administrator is necessary to grant the managed identity on deployIfNotExists or modify assignments necessary permissions. All policy objects will be readable to all roles over the scope.

If none of the Built-in roles have the permissions required, create a custom role.

**Note:** The managed identity of a deployIfNotExists or modify policy assignment needs enough permissions to create or update targetted resources.

## Resources covered by Azure Policy

Azure Policy evaluates all Azure resources at or below subscription-level, including Arc enabled resources. For certain resource providers such as guest configuration, Azure Kubernetes Service, and Azure Key Vault, there's a deeper integration for managing settings and objects.

## Recommendations for managing policies

Here are a few pointers and tips to keep in mind:

- Start with an audit effect instead of a deny effect to track impact of your policy definition on the resources in your environment. If you have scripts already in place to autoscale your applications, setting a deny effect may hinder such automation tasks already in place.
- Consider organizational hierarchies when creating definitions and assignments. We recommend creating definitions at higher levels such as the management group or subscription level. Then, create the assignment at the next child level. If you create a definition at a management group, the assignment can be scoped down to a subscription or resource group within that management group.
- We recommend creating and assigning initiative definitions even for a single policy definition. For example, you have policy definition policyDefA and create it under initiative definition initiativeDefC. If you create another policy definition later for policyDefB with goals similar to policyDefA, you can add it under initiativeDefC and track them together.
- Once you've created an initiative assignment, policy definitions added to the initiative also become part of that initiative's assignments.
- When an initiative assignment is evaluated, all policies within the initiative are also evaluated. If you need to evaluate a policy individually, it's better to not include it in an initiative.

## Azure Policy objects

### Policy definition

The journey of creating and implementing a policy in Azure Policy begins with creating a policy definition. Every policy definition has conditions under which it's enforced. And, it has a defined effect that takes place if the conditions are met.

In Azure Policy, we offer several built-in policies that are available by default. For example:

- Allowed Storage Account SKUs (Deny): Determines if a storage account being deployed is within a set of SKU sizes. Its effect is to deny all storage accounts that don't adhere to the set of defined SKU sizes.

- Allowed Resource Type (Deny): Defines the resource types that you can deploy. Its effect is to deny all resources that aren't part of this defined list.
- Allowed Locations (Deny): Restricts the available locations for new resources. Its effect is used to enforce your geo-compliance requirements.
- Allowed Virtual Machine SKUs (Deny): Specifies a set of virtual machine SKUs that you can deploy.
- Add a tag to resources (Modify): Applies a required tag and its default value if it's not specified by the deploy request.
- Not allowed resource types (Deny): Prevents a list of resource types from being deployed.

To implement these policy definitions (both built-in and custom definitions), you'll need to assign them. You can assign any of these policies through the Azure portal, PowerShell, or Azure CLI.

Policy evaluation happens with several different actions, such as policy assignment or policy updates. Policy parameters help simplify your policy management by reducing the number of policy definitions you must create. You can define parameters when creating a policy definition to make it more generic. Then you can reuse that policy definition for different scenarios. You do so by passing in different values when assigning the policy definition. For example, specifying one set of locations for a subscription.

Parameters are defined when creating a policy definition. When a parameter is defined, it's given a name and optionally given a value. For example, you could define a parameter for a policy titled location. Then you can give it different values such as EastUS or WestUS when assigning a policy.

## Initiative definition

An initiative definition is a collection of policy definitions that are tailored toward achieving a singular overarching goal. Initiative definitions simplify managing and assigning policy definitions. They simplify by grouping a set of policies as one single item. For example, you could create an initiative titled Enable Monitoring in Azure Security Center, with a goal to monitor all the available security recommendations in your Azure Security Center.

**Note:** The SDK, such as Azure CLI and Azure PowerShell, use properties and parameters named PolicySet to refer to initiatives.

Under this initiative, you would have policy definitions such as:

- Monitor unencrypted SQL Database in Security Center - For monitoring unencrypted SQL databases and servers.
- Monitor OS vulnerabilities in Security Center - For monitoring servers that don't satisfy the configured baseline.
- Monitor missing Endpoint Protection in Security Center - For monitoring servers without an installed endpoint protection agent.

Like policy parameters, initiative parameters help simplify initiative management by reducing redundancy. Initiative parameters are parameters being used by the policy definitions within the initiative.

For example, take a scenario where you have an initiative definition - initiativeC, with policy definitions policyA and policyB each expecting a different type of parameter:

Policy	Name of parameter	Type of parameter	Note
policyA	allowedLocations	array	This parameter expects a list of strings for a value since the parameter type has been defined as an array
policyB	allowedSingleLocation	string	This parameter expects one word for a value since the parameter type has been defined as a string

In this scenario, when defining the initiative parameters for initiativeC, you have three options:

- Use the parameters of the policy definitions within this initiative: In this example, allowedLocations and allowedSingleLocation become initiative parameters for initiativeC.
- Provide values to the parameters of the policy definitions within this initiative definition. In this example, you can provide a list of locations to policyA's parameter - allowedLocations and policyB's parameter - allowedSingleLocation. You can also provide values when assigning this initiative.
- Provide a list of value options that can be used when assigning this initiative. When you assign this initiative, the inherited parameters from the policy definitions within the initiative, can only have values from this provided list.

When creating value options in an initiative definition, you're unable to input a different value during the initiative assignment because it's not part of the list.

## Assignments

An assignment is a policy definition or initiative that has been assigned to take place within a specific scope. This scope could range from a management group to an individual resource. The term scope refers to all the resources, resource groups, subscriptions, or management groups that the definition is assigned to. Assignments are inherited by all child resources. This design means that a definition applied to a resource group is also applied to resources in that resource group. However, you can exclude a subscope from the assignment.

For example, at the subscription scope, you can assign a definition that prevents the creation of networking resources. You could exclude a resource group in that subscription

that is intended for networking infrastructure. You then grant access to this networking resource group to users that you trust with creating networking resources.

In another example, you might want to assign a resource type allowlist definition at the management group level. Then you assign a more permissive policy (allowing more resource types) on a child management group or even directly on subscriptions. However, this example wouldn't work because Azure Policy is an explicit deny system. Instead, you need to exclude the child management group or subscription from the management group-level assignment. Then, assign the more permissive definition on the child management group or subscription level. If any assignment results in a resource getting denied, then the only way to allow the resource is to modify the denying assignment.

## Maximum count of Azure Policy objects

There's a maximum count for each object type for Azure Policy. For definitions, an entry of Scope means the management group or subscription. For assignments and exemptions, an entry of Scope means the management group, subscription, resource group, or individual resource.

Where	What	Maximum count
Scope	Policy definitions	500
Scope	Initiative definitions	200
Tenant	Initiative definitions	2,500
Scope	Policy or initiative assignments	200
Scope	Exemptions	1000
Policy definition	Parameters	20

Initiative definition	Policies	1000
Initiative definition	Parameters	300
Policy or initiative assignments	Exclusions (notScopes)	400
Policy rule	Nested conditionals	512
Remediation task	Resources	500

## Azure Policy definition structure

Azure Policy establishes conventions for resources. Policy definitions describe resource compliance conditions and the effect to take if a condition is met. A condition compares a resource property field or a value to a required value. Resource property fields are accessed by using aliases. When a resource property field is an array, a special array alias can be used to select values from all array members and apply a condition to each one.

By defining conventions, you can control costs and more easily manage your resources. For example, you can specify that only certain types of virtual machines are allowed. Or, you can require that resources have a particular tag. Policy assignments are inherited by child resources. If a policy assignment is applied to a resource group, it's applicable to all the resources in that resource group.

You use JSON to create a policy definition. The policy definition contains elements for:

- display name
- description
- mode
- metadata
- parameters
- policy rule
  - logical evaluation
  - effect

For example, the following JSON shows a policy that limits where resources are deployed:

```
{  
  "properties": {  
    "displayName": "Allowed locations",  
    "description": "This policy enables you to restrict the locations your organization can specify when deploying resources.",  
    "mode": "Indexed",  
    "metadata": {  
      "version": "1.0.0",  
      "category": "Locations"  
    },  
    "parameters": {  
      "allowedLocations": {  
        "type": "array",  
        "metadata": {  
          "description": "The list of locations that can be specified when deploying resources",  
          "strongType": "location",  
          "displayName": "Allowed locations"  
        },  
        "defaultValue": [ "westus2" ]  
      }  
    },  
    "policyRule": {  
      "if": {  
        "not": {  
          "field": "location",  
          "in": "[parameters('allowedLocations')]"  
        }  
      },  
      "then": {  
        "effect": "deny"  
      }  
    }  
  }  
}
```

## Display name and description

You use displayName and description to identify the policy definition and provide context for when it's used. displayName has a maximum length of 128 characters and description a maximum length of 512 characters.

## Type

While the type property can't be set, there are three values that are returned by SDK and visible in the portal:

- **Builtin:** These policy definitions are provided and maintained by Microsoft.
- **Custom:** All policy definitions created by customers have this value.
- **Static:** Indicates a Regulatory Compliance policy definition with Microsoft Ownership. The compliance results for these policy definitions are the results of third-party audits on Microsoft infrastructure. In the Azure portal, this value is sometimes displayed as Microsoft managed.

## Mode

Mode is configured depending on if the policy is targeting an Azure Resource Manager property or a Resource Provider property.

### Resource Manager modes

The mode determines which resource types are evaluated for a policy definition. The supported modes are:

- **all:** evaluate resource groups, subscriptions, and all resource types
- **indexed:** only evaluate resource types that support tags and location

For example, resource Microsoft.Network/routeTables supports tags and location and is evaluated in both modes. However, resource Microsoft.Network/routeTables/routes can't be tagged and isn't evaluated in Indexed mode.

We recommend that you set mode to all in most cases. All policy definitions created through the portal use the all mode. If you use PowerShell or Azure CLI, you can specify the mode parameter manually. If the policy definition doesn't include a mode value, it defaults to all in Azure PowerShell and to null in Azure CLI. A null mode is the same as using indexed to support backward compatibility.

Indexed should be used when creating policies that enforce tags or locations. While not required, it prevents resources that don't support tags and locations from showing up as non-compliant in the compliance results. The exception is resource groups and subscriptions. Policy definitions that enforce location or tags on a resource group or subscription should set mode to all and specifically target the Microsoft.Resources/subscriptions/resourceGroups or Microsoft.Resources/subscriptions type.

## Resource Provider modes

The following Resource Provider modes are fully supported:

- Microsoft.Kubernetes.Data for managing your Kubernetes clusters on or off Azure. Definitions using this Resource Provider mode use effects audit, deny, and disabled. This mode supports custom definitions as a public preview.
- Microsoft.KeyVault.Data for managing vaults and certificates in Azure Key Vault.

The following Resource Provider mode is currently supported as a preview:

- Microsoft.ContainerService.Data for managing admission controller rules on Azure Kubernetes Service. Definitions using this Resource Provider mode must use the EnforceRegoPolicy effect. This mode is deprecated.

**Note:** Unless explicitly stated, Resource Provider modes only support built-in policy definitions, and exemptions are not supported at the component-level.

## Metadata

The optional metadata property stores information about the policy definition. Customers can define any properties and values useful to their organization in metadata. However, there are some common properties used by Azure Policy and in built-ins. Each metadata property has a limit of 1024 characters.

### Common metadata properties

- version (string): Tracks details about the version of the contents of a policy definition.
- category (string): Determines under which category in the Azure portal the policy definition is displayed.
- preview (boolean): True or false flag for if the policy definition is preview.
- deprecated (boolean): True or false flag for if the policy definition has been marked as deprecated.
- portalReview (string): Determines whether parameters should be reviewed in the portal, regardless of the required input.

**Note:** The Azure Policy service uses version, preview, and deprecated properties to convey level of change to a built-in policy definition or initiative and state. The format of version is: {Major}.{Minor}.{Patch}. Specific states, such as deprecated or preview, are appended to the version property or in another property as a boolean.

## Parameters

Parameters help simplify your policy management by reducing the number of policy definitions. Think of parameters like the fields on a form - name, address, city, state. These parameters always stay the same, however their values change based on the individual filling out the form. Parameters work the same way when building policies. By including parameters in a policy definition, you can reuse that policy for different scenarios by using different values.

**Note:** Parameters may be added to an existing and assigned definition. The new parameter must include the defaultValue property. This prevents existing assignments of the policy or initiative from indirectly being made invalid.

## Parameter properties

A parameter has the following properties that are used in the policy definition:

- name: The name of your parameter. Used by the parameters deployment function within the policy rule.
  - type: Determines if the parameter is a string, array, object, boolean, integer, float, or datetime.
  - metadata: Defines subproperties primarily used by the Azure portal to display user-friendly information:
    - description: The explanation of what the parameter is used for. Can be used to provide examples of acceptable values.
    - displayName: The friendly name shown in the portal for the parameter.
    - strongType: (Optional) Used when assigning the policy definition through the portal. Provides a context aware list.
    - assignPermissions: (Optional) Set as true to have Azure portal create role assignments during policy assignment. This property is useful in case you wish to assign permissions outside the assignment scope. There's one role assignment per role definition in the policy (or per role definition in all of the policies in the initiative). The parameter value must be a valid resource or scope.
  - defaultValue: (Optional) Sets the value of the parameter in an assignment if no value is given. Required when updating an existing policy definition that is assigned.
  - allowedValues: (Optional) Provides an array of values that the parameter accepts during assignment. Allowed value comparisons are case-sensitive.

As an example, you could define a policy definition to limit the locations where resources can be deployed. A parameter for that policy definition could be allowedLocations. This parameter would be used by each assignment of the policy definition to limit the accepted values. The use of strongType provides an enhanced experience when completing the assignment through the portal:

```

    "eastus2",
    "westus2",
    "westus"
]
}
}

```

## Using a parameter value

In the policy rule, you reference parameters with the following parameters function syntax:

```
{
  "field": "location",
  "in": "[parameters('allowedLocations')]"
}
```

## strongType

Within the metadata property, you can use strongType to provide a multiselect list of options within the Azure portal. strongType can be a supported resource type or an allowed value. To determine whether a resource type is valid for strongType, use [Get-AzResourceProvider](#). The format for a resource type strongType is <Resource Provider>/<Resource Type>. For example, Microsoft.Network/virtualNetworks/subnets.

Some resource types not returned by Get-AzResourceProvider are supported. Those types are:

- Microsoft.RecoveryServices/vaults/backupPolicies

The non resource type allowed values for strongType are:

- location
- resourceTypes
- storageSkus
- vmSKUs
- existingResourceGroups

## Definition location

While creating an initiative or policy, it's necessary to specify the definition location. The definition location must be a management group or a subscription. This location determines the scope to which the initiative or policy can be assigned. Resources must be direct members of or children within the hierarchy of the definition location to target for assignment.

If the definition location is a:

- Subscription - Only resources within that subscription can be assigned the policy definition.
- Management group - Only resources within child management groups and child subscriptions can be assigned the policy definition. If you plan to apply the policy definition to several subscriptions, the location must be a management group that contains each subscription.

## Policy rule

The policy rule consists of If and Then blocks. In the If block, you define one or more conditions that specify when the policy is enforced. You can apply logical operators to these conditions to precisely define the scenario for a policy.

In the Then block, you define the effect that happens when the If conditions are fulfilled.

```
{  
  "if": {  
    <condition> | <logical operator>  
  },  
  "then": {  
    "effect": "deny | audit | modify | append | auditIfNotExists | deployIfNotExists | disabled"  
  }  
}
```

## Logical operators

Supported logical operators are:

- "not": {condition or operator}
- "allOf": [{condition or operator},{condition or operator}]
- "anyOf": [{condition or operator},{condition or operator}]

The not syntax inverts the result of the condition. The allOf syntax (similar to the logical And operation) requires all conditions to be true. The anyOf syntax (similar to the logical Or operation) requires one or more conditions to be true.

You can nest logical operators. The following example shows a not operation that is nested within an allOf operation.

```
"if": {  
  "allOf": [{  
    "not": {  
      "field": "tags",  
      "containsKey": "application"  
    }  
  },  
}
```

```

    {
      "field": "type",
      "equals": "Microsoft.Storage/storageAccounts"
    }
  ]
},

```

## Conditions

A condition evaluates whether a value meets certain criteria. The supported conditions are:

- "equals": "stringValue"
- "notEquals": "stringValue"
- "like": "stringValue"
- "notLike": "stringValue"
- "match": "stringValue"
- "matchInsensitively": "stringValue"
- "notMatch": "stringValue"
- "notMatchInsensitively": "stringValue"
- "contains": "stringValue"
- "notContains": "stringValue"
- "in": ["stringValue1", "stringValue2"]
- "notIn": ["stringValue1", "stringValue2"]
- "containsKey": "keyName"
- "notContainsKey": "keyName"
- "less": "dateValue" | "less": "stringValue" | "less": intValue
- "lessOrEquals": "dateValue" | "lessOrEquals": "stringValue" | "lessOrEquals": intValue
- "greater": "dateValue" | "greater": "stringValue" | "greater": intValue
- "greaterOrEquals": "dateValue" | "greaterOrEquals": "stringValue" | "greaterOrEquals": intValue
- "exists": "bool"

For less, lessOrEquals, greater, and greaterOrEquals, if the property type doesn't match the condition type, an error is thrown. String comparisons are made using InvariantCultureIgnoreCase.

When using the like and notLike conditions, you provide a wildcard \* in the value. The value shouldn't have more than one wildcard \*.

When using the match and notMatch conditions, provide # to match a digit, ? for a letter, . to match any character, and any other character to match that actual character. While match and notMatch are case-sensitive, all other conditions that evaluate a stringValue are case-insensitive. Case-insensitive alternatives are available in matchInsensitively and notMatchInsensitively.

## Fields

Conditions that evaluate whether the values of properties in the resource request payload meet certain criteria can be formed using a field expression. The following fields are supported:

- name
- fullName
  - Returns the full name of the resource. The full name of a resource is the resource name prepended by any parent resource names (for example "myServer/myDatabase").
- kind
- type
- location
  - Location fields are normalized to support various formats. For example, East US 2 is considered equal to eastus2.
  - Use global for resources that are location agnostic.
- id
  - Returns the resource ID of the resource that is being evaluated.
  - Example: /subscriptions/06be863d-0996-4d56-be22-384767287aa2/resourceGroups/myRG/providers/Microsoft.KeyVault/vaults/myVault
- identity.type
  - Returns the type of managed identity enabled on the resource.
- tags
- tags['<tagName>']
  - This bracket syntax supports tag names that have punctuation such as a hyphen, period, or space.
  - Where <tagName> is the name of the tag to validate the condition for.
  - Examples: tags['Acct.CostCenter'] where Acct.CostCenter is the name of the tag.
- tags["<tagName>"]
  - This bracket syntax supports tag names that have apostrophes in it by escaping with double apostrophes.
  - Where '<tagName>' is the name of the tag to validate the condition for.
  - Example: tags["'My.Apostrophe.Tag'"] where 'My.Apostrophe.Tag' is the name of the tag.

**Note:** tags.<tagName>, tags[tagName], and tags[tag.with.dots] are still acceptable ways of declaring a tags field. However, the preferred expressions are those listed above.

**Note:** In field expressions referring to [\*] alias, each element in the array is evaluated individually with logical and between elements.

## Use tags with parameters

A parameter value can be passed to a tag field. Passing a parameter to a tag field increases the flexibility of the policy definition during policy assignment.

In the following example, concat is used to create a tags field lookup for the tag named the value of the tagName parameter. If that tag doesn't exist, the modify effect is used to add the tag using the value of the same named tag set on the audited resources parent resource group by using the resourcegroup() lookup function.

```
{  
  "if": {  
    "field": "[concat('tags[', parameters('tagName'), ']')]",  
    "exists": "false"  
  },  
  "then": {  
    "effect": "modify",  
    "details": {  
      "operations": [{  
        "operation": "add",  
        "field": "[concat('tags[', parameters('tagName'), ']')]",  
        "value": "[resourcegroup().tags[parameters('tagName')]]"  
      }],  
      "roleDefinitionIds": [  
        "/providers/microsoft.authorization/roleDefinitions/  
4a9ae827-6dc8-4573-8ac7-8239d42aa03f"  
      ]  
    }  
  }  
}
```

## Value

Conditions that evaluate whether a value meets certain criteria can be formed using a value expression. Values can be literals, the values of parameters, or the returned values of any supported template functions.

If the result of a template function is an error, policy evaluation fails. A failed evaluation is an implicit deny.

## Value examples

This policy rule example uses value to compare the result of the resourceGroup() function and the returned name property to a like condition of \*netrg. The rule denies any resource not of the Microsoft.Network/\* type in any resource group whose name ends in \*netrg.

```
{
  "if": {
    "allOf": [
      {"value": "[resourceGroup().name]",
       "like": "*netrg"
      },
      {
        "field": "type",
        "notLike": "Microsoft.Network/*"
      }
    ]
  },
  "then": {
    "effect": "deny"
  }
}
```

This policy rule example uses value to check if the result of multiple nested functions equals true. The rule denies any resource that doesn't have at least three tags.

```
{
  "mode": "indexed",
  "policyRule": {
    "if": {
      "value": "[less(length(field('tags')), 3)]",
      "equals": "true"
    },
    "then": {
      "effect": "deny"
    }
  }
}
```

## Avoiding template failures

The use of template functions in value allows for many complex nested functions. If the result of a template function is an error, policy evaluation fails. A failed evaluation is an implicit deny. An example of a value that fails in certain scenarios:

```
{
  "policyRule": {
    "if": {
      "value": "[substring(field('name'), 0, 3)]",
      "equals": "abc"
    }
  }
}
```

```

    },
    "then": {
        "effect": "audit"
    }
}
}

```

The example policy rule above uses substring() to compare the first three characters of name to abc. If name is shorter than three characters, the substring() function results in an error. This error causes the policy to become a deny effect.

Instead, use the if() function to check if the first three characters of name equal abc without allowing a name shorter than three characters to cause an error:

JSON

Copy

```
{
    "policyRule": {
        "if": {
            "value": "[if(greaterOrEquals(length(field('name')), 3), substring(field('name'), 0, 3), 'not starting with abc')]",
            "equals": "abc"
        },
        "then": {
            "effect": "audit"
        }
    }
}
```

With the revised policy rule, if() checks the length of name before trying to get a substring() on a value with fewer than three characters. If name is too short, the value "not starting with abc" is returned instead and compared to abc. A resource with a short name that doesn't begin with abc still fails the policy rule, but no longer causes an error during evaluation.

## Count

Conditions that count how many members of an array meet certain criteria can be formed using a count expression. Common scenarios are checking whether 'at least one of', 'exactly one of', 'all of', or 'none of' the array members satisfy a condition. Count evaluates each array member for a condition expression and sums the true results, which is then compared to the expression operator.

## Field count

Count how many members of an array in the request payload satisfy a condition expression. The structure of field count expressions is:

```
{  
  "count": {  
    "field": "<[*] alias>",  
    "where": {  
      /* condition expression */  
    }  
  },  
  "<condition>": "<compare the count of true condition expression array members to this  
value>"  
}
```

The following properties are used with field count:

- count.field (required): Contains the path to the array and must be an array alias.
- count.where (optional): The condition expression to individually evaluate for each [\*] alias array member of count.field. If this property isn't provided, all array members with the path of 'field' are evaluated to true. Any condition can be used inside this property. Logical operators can be used inside this property to create complex evaluation requirements.
- <condition> (required): The value is compared to the number of items that met the count.where condition expression. A numeric condition should be used.

Field count expressions can enumerate the same field array up to three times in a single policyRule definition.

## Value count

Count how many members of an array satisfy a condition. The array can be a literal array or a reference to array parameter. The structure of value count expressions is:

```
{  
  "count": {  
    "value": "<literal array | array parameter reference>",  
    "name": "<index name>",  
    "where": {  
      /* condition expression */  
    }  
  },  
  "<condition>": "<compare the count of true condition expression array members to this  
value>"  
}
```

The following properties are used with value count:

- `count.value` (required): The array to evaluate.
- `count.name` (required): The index name, composed of English letters and digits. Defines a name for the value of the array member evaluated in the current iteration. The name is used for referencing the current value inside the `count.where` condition. Optional when the `count` expression isn't in a child of another `count` expression. When not provided, the index name is implicitly set to "default".
- `count.where` (optional): The condition expression to individually evaluate for each array member of `count.value`. If this property isn't provided, all array members are evaluated to true. Any condition can be used inside this property. Logical operators can be used inside this property to create complex evaluation requirements. The value of the currently enumerated array member can be accessed by calling the `current` function.
- `<condition>` (required): The value is compared to the number of items that met the `count.where` condition expression. A numeric condition should be used.

The following limits are enforced:

- Up to 10 value count expressions can be used in a single `policyRule` definition.
- Each value count expression can perform up to 100 iterations. This number includes the number of iterations performed by any parent value count expressions.

## The current function

The `current()` function is only available inside the `count.where` condition. It returns the value of the array member that is currently enumerated by the `count` expression evaluation.

### *Value count usage*

- `current(<index name defined in count.name>)`. For example: `current('arrayMember')`.
- `current()`. Allowed only when the value count expression isn't a child of another `count` expression. Returns the same value as above.

If the value returned by the call is an object, property accessors are supported. For example: `current('objectArrayMember').property`.

### *Field count usage*

- `current(<the array alias defined in count.field>)`. For example, `current('Microsoft.Test/resource/enumeratedArray[*]')`.
- `current()`. Allowed only when the field count expression isn't a child of another `count` expression. Returns the same value as above.
- `current(<alias of a property of the array member>)`. For example, `current('Microsoft.Test/resource/enumeratedArray[*].property')`.

## Field count examples

Example 1: Check if an array is empty

```
{  
  "count": {
```

```
        "field": "Microsoft.Network/networkSecurityGroups/securityRules[*]"  
    },  
    "equals": 0  
}
```

Example 2: Check for only one array member to meet the condition expression

```
{  
    "count": {  
        "field": "Microsoft.Network/networkSecurityGroups/securityRules[*]",  
        "where": {  
            "field": "Microsoft.Network/networkSecurityGroups/securityRules[*].description",  
            "equals": "My unique description"  
        }  
    },  
    "equals": 1  
}
```

Example 3: Check for at least one array member to meet the condition expression

```
{  
    "count": {  
        "field": "Microsoft.Network/networkSecurityGroups/securityRules[*]",  
        "where": {  
            "field": "Microsoft.Network/networkSecurityGroups/securityRules[*].description",  
            "equals": "My common description"  
        }  
    },  
    "greaterOrEquals": 1  
}
```

Example 4: Check that all object array members meet the condition expression

```
{  
    "count": {  
        "field": "Microsoft.Network/networkSecurityGroups/securityRules[*]",  
        "where": {  
            "field": "Microsoft.Network/networkSecurityGroups/securityRules[*].description",  
            "equals": "description"  
        }  
    },  
    "equals": "[length(field('Microsoft.Network/networkSecurityGroups/securityRules[*']))]"  
}
```

Example 5: Check that at least one array member matches multiple properties in the condition expression

```
{  
    "count": {  
        "field": "Microsoft.Network/networkSecurityGroups/securityRules[*]",  
        "where": {  
            "field": "Microsoft.Network/networkSecurityGroups/securityRules[*].name",  
            "equals": "Allow-HTTP"  
        }  
    },  
    "greaterOrEquals": 1  
}
```

```

"where": {
  "allOf": [
    {
      "field": "Microsoft.Network/networkSecurityGroups/securityRules[*].direction",
      "equals": "Inbound"
    },
    {
      "field": "Microsoft.Network/networkSecurityGroups/securityRules[*].access",
      "equals": "Allow"
    },
    {
      "field": "Microsoft.Network/networkSecurityGroups/
securityRules[*].destinationPortRange",
      "equals": "3389"
    }
  ]
},
"greater": 0
}

```

Example 6: Use current() function inside the where conditions to access the value of the currently enumerated array member in a template function. This condition checks whether a virtual network contains an address prefix that isn't under the 10.0.0.0/24 CIDR range.

```

{
  "count": {
    "field": "Microsoft.Network/virtualNetworks/addressSpace.addressPrefixes[*]",
    "where": {
      "value": "[ipRangeContains('10.0.0.0/24', current('Microsoft.Network/virtualNetworks/
addressSpace.addressPrefixes[*]'))]",
      "equals": false
    }
  },
  "greater": 0
}

```

Example 7: Use field() function inside the where conditions to access the value of the currently enumerated array member. This condition checks whether a virtual network contains an address prefix that isn't under the 10.0.0.0/24 CIDR range.

```

{
  "count": {
    "field": "Microsoft.Network/virtualNetworks/addressSpace.addressPrefixes[*]",

```

```

    "where": {
        "value": "[ipRangeContains('10.0.0.0/24', first(field(('Microsoft.Network/virtualNetworks/addressSpace.addressPrefixes[*]'))))]",
        "equals": false
    },
    "greater": 0
}

```

## Value count examples

Example 1: Check if resource name matches any of the given name patterns.

```
{
    "count": {
        "value": [ "prefix1_*", "prefix2_*" ],
        "name": "pattern",
        "where": {
            "field": "name",
            "like": "[current('pattern')]"
        }
    },
    "greater": 0
}
```

Example 2: Check if resource name matches any of the given name patterns. The current() function doesn't specify an index name. The outcome is the same as the previous example.

```
{
    "count": {
        "value": [ "prefix1_*", "prefix2_*" ],
        "where": {
            "field": "name",
            "like": "[current()]"
        }
    },
    "greater": 0
}
```

Example 3: Check if resource name matches any of the given name patterns provided by an array parameter.

```
{
    "count": {
        "value": "[parameters('namePatterns')]",
```

```

    "name": "pattern",
    "where": {
        "field": "name",
        "like": "[current('pattern')]"
    },
    "greater": 0
}

```

Example 4: Check if any of the virtual network address prefixes isn't under the list of approved prefixes.

```

{
    "count": {
        "field": "Microsoft.Network/virtualNetworks/addressSpace.addressPrefixes[*]",
        "where": {
            "count": {
                "value": "[parameters('approvedPrefixes')]",
                "name": "approvedPrefix",
                "where": {
                    "value": "[ipRangeContains(current('approvedPrefix'), current('Microsoft.Network/virtualNetworks/addressSpace.addressPrefixes[*]'))]",
                    "equals": true
                },
                "},
                "equals": 0
            }
        },
        "greater": 0
}

```

Example 5: Check that all the reserved NSG rules are defined in an NSG. The properties of the reserved NSG rules are defined in an array parameter containing objects.

Parameter value:

```
[
    {
        "priority": 101,
        "access": "deny",
        "direction": "inbound",
        "destinationPortRange": 22
    },
    {
        "priority": 102,
        "access": "deny",

```

```

    "direction": "inbound",
    "destinationPortRange": 3389
  }
]

```

Policy:

```

{
  "count": {
    "value": "[parameters('reservedNsgRules')]",
    "name": "reservedNsgRule",
    "where": {
      "count": {
        "field": "Microsoft.Network/networkSecurityGroups/securityRules[*]",
        "where": {
          "allOf": [
            {
              "field": "Microsoft.Network/networkSecurityGroups/
securityRules[*].priority",
              "equals": "[current('reservedNsgRule').priority]"
            },
            {
              "field": "Microsoft.Network/networkSecurityGroups/
securityRules[*].access",
              "equals": "[current('reservedNsgRule').access]"
            },
            {
              "field": "Microsoft.Network/networkSecurityGroups/
securityRules[*].direction",
              "equals": "[current('reservedNsgRule').direction]"
            },
            {
              "field": "Microsoft.Network/networkSecurityGroups/
securityRules[*].destinationPortRange",
              "equals": "[current('reservedNsgRule').destinationPortRange]"
            }
          ]
        }
      },
      "equals": 1
    }
  }
},

```

```
"equals": "[length(parameters('reservedNsgRules'))]"  
}
```

## Effect

Azure Policy supports the following types of effect:

- Append: adds the defined set of fields to the request
- Audit: generates a warning event in activity log but doesn't fail the request
- AuditIfNotExists: generates a warning event in activity log if a related resource doesn't exist
- Deny: generates an event in the activity log and fails the request
- DeployIfNotExists: deploys a related resource if it doesn't already exist
- Disabled: doesn't evaluate resources for compliance to the policy rule
- Modify: adds, updates, or removes the defined tags from a resource or subscription
- EnforceOPAConstraint (deprecated): configures the Open Policy Agent admissions controller with Gatekeeper v3 for self-managed Kubernetes clusters on Azure
- EnforceRegoPolicy (deprecated): configures the Open Policy Agent admissions controller with Gatekeeper v2 in Azure Kubernetes Service

## Policy functions

Functions can be used to introduce additional logic into a policy rule. They are resolved within the policy rule of a policy definition and within parameter values assigned to policy definitions in an initiative.

All Resource Manager template functions are available to use within a policy rule, except the following functions and user-defined functions:

- copyIndex()
- deployment()
- list\*
- newGuid()
- pickZones()
- providers()
- reference()
- resourceId()
- variables()

**Note:** These functions are still available within the details.deployment.properties.template portion of the template deployment in a deployIfNotExists policy definition.

The following function is available to use in a policy rule, but differs from use in an Azure Resource Manager template (ARM template):

- `utcNow()` - Unlike an ARM template, this property can be used outside `defaultValue`.
  - Returns a string that is set to the current date and time in Universal ISO 8601 `DateTime` format `yyyy-MM-ddTHH:mm:ss.fffffffZ`.

The following functions are only available in policy rules:

- `addDays(dateTime, numberOfDaysToAdd)`
  - `dateTime`: [Required] string - String in the Universal ISO 8601 `DateTime` format '`yyyy-MM-ddTHH:mm:ss.FFFFFFFFZ`'
  - `numberOfDaysToAdd`: [Required] integer - Number of days to add
- `field(fieldName)`
  - `fieldName`: [Required] string - Name of the field to retrieve
  - Returns the value of that field from the resource that is being evaluated by the `If` condition.
  - `field` is primarily used with `AuditIfNotExists` and `DeployIfNotExists` to reference fields on the resource that are being evaluated. An example of this use can be seen in the `DeployIfNotExists` example.
- `requestContext().apiVersion`
  - Returns the API version of the request that triggered policy evaluation (example: `2021-09-01`). This value is the API version that was used in the `PUT/PATCH` request for evaluations on resource creation/update. The latest API version is always used during compliance evaluation on existing resources.
- `policy()`
  - Returns the following information about the policy that is being evaluated. Properties can be accessed from the returned object (example: `[policy().assignmentId]`).

```
{  
  "assignmentId": "/subscriptions/ad404ddd-36a5-4ea8-b3e3-681e77487a63/providers/  
Microsoft.Authorization/policyAssignments/myAssignment",  
  "definitionId": "/providers/Microsoft.Authorization/policyDefinitions/  
34c877ad-507e-4c82-993e-3452a6e0ad3c",  
  "setDefinitionId": "/providers/Microsoft.Authorization/policySetDefinitions/  
42a694ed-f65e-42b2-aa9e-8052e9740a92",  
  "definitionReferenceId": "StorageAccountNetworkACLs"  
}
```

- `ipRangeContains(range, targetRange)`
  - `range`: [Required] string - String specifying a range of IP addresses to check if the `targetRange` is within.
  - `targetRange`: [Required] string - String specifying a range of IP addresses to validate as included within the range.

- Returns a boolean for whether the range IP address range contains the targetRange IP address range. Empty ranges, or mixing between IP families isn't allowed and results in evaluation failure.
- Supported formats:
  - Single IP address (examples: 10.0.0.0, 2001:0DB8::3:FFFE)
  - CIDR range (examples: 10.0.0.0/24, 2001:0DB8::/110)
  - Range defined by start and end IP addresses (examples: 192.168.0.1-192.168.0.9, 2001:0DB8::2001:0DB8::3:FFFF)
- current(indexName)
  - Special function that may only be used inside count expressions.

### Policy function example

This policy rule example uses the resourceGroup resource function to get the name property, combined with the concat array and object function to build a like condition that enforces the resource name to start with the resource group name.

```
{
  "if": {
    "not": {
      "field": "name",
      "like": "[concat(resourceGroup().name,'*')]"
    }
  },
  "then": {
    "effect": "deny"
  }
}
```

## Aliases

You use property aliases to access specific properties for a resource type. Aliases enable you to restrict what values or conditions are allowed for a property on a resource. Each alias maps to paths in different API versions for a given resource type. During policy evaluation, the policy engine gets the property path for that API version.

The list of aliases is always growing. To find what aliases are currently supported by Azure Policy, use one of the following methods:

- Azure Policy extension for Visual Studio Code (recommended)  
Use the Azure Policy extension for Visual Studio Code to view and discover aliases

for resource properties.

```
    "hardwareProfile": {  
      "vmSize": "Standard_D2"  
    },  
    "storageProfile": {  
      "imageReference": {  
        "publisher": "Copy",  
        "offer": "Windows-10",  
        "sku": "rs5-pro",  
        "version": "latest",  
      }  
    }  
  }  
}  
Microsoft.Compute/imageOffer Microsoft.Compute/virtualMachines/imageOffer  
Microsoft.Compute/virtualMachines/storageProfile.imageReference.offer
```

- Azure PowerShell

```
# Login first with Connect-AzAccount if not using Cloud Shell
```

```
# Use Get-AzPolicyAlias to list available providers
```

```
Get-AzPolicyAlias -ListAvailable
```

```
# Use Get-AzPolicyAlias to list aliases for a Namespace (such as Azure Compute --  
Microsoft.Compute)
```

```
(Get-AzPolicyAlias -NamespaceMatch 'compute').Aliases
```

To find aliases that can be used with the modify effect, use the following command in Azure PowerShell 4.6.0 or higher:

```
Get-AzPolicyAlias | Select-Object -ExpandProperty 'Aliases' | Where-Object  
{ $_.DefaultMetadata.Attributes -eq 'Modifiable' }
```

- Azure CLI

```
# Login first with az login if not using Cloud Shell
```

```
# List namespaces
```

```
az provider list --query [*].namespace
```

```
# Get Azure Policy aliases for a specific Namespace (such as Azure Compute --  
Microsoft.Compute)
```

```
az provider show --namespace Microsoft.Compute --expand "resourceTypes/aliases"  
--query "resourceTypes[].aliases[].name"
```

- REST API / ARMClient

```
G E T  h t t p s : / / m a n a g e m e n t . a z u r e . c o m / p r o v i d e r s / ? a p i -  
v e r s i o n = 2 0 1 9 - 1 0 - 0 1 & $ e x p a n d = r e s o u r c e T y p e s / a l i a s e s
```

## Understanding the [\*] alias

Several of the aliases that are available have a version that appears as a 'normal' name and another that has [\*] attached to it. For example:

- Microsoft.Storage/storageAccounts/networkAcls.ipRules
- Microsoft.Storage/storageAccounts/networkAcls.ipRules[\*]

The 'normal' alias represents the field as a single value. This field is for exact match comparison scenarios when the entire set of values must be exactly as defined, no more and no less.

The [\*] alias represents a collection of values selected from the elements of an array resource property. For example:

Alias	Selected values
Microsoft.Storage/storageAccounts/networkAcls.ipRules[*]	The elements of the ipRules array.
Microsoft.Storage/storageAccounts/networkAcls.ipRules[*].action	The values of the action property from each element of the ipRules array.

When used in a field condition, array aliases make it possible to compare each individual array element to a target value. When used with count expression, it's possible to:

- Check the size of an array
- Check if all\any\none of the array elements meet a complex condition
- Check if exactly n array elements meet a complex condition

# Azure Portal

## What is the Azure portal?

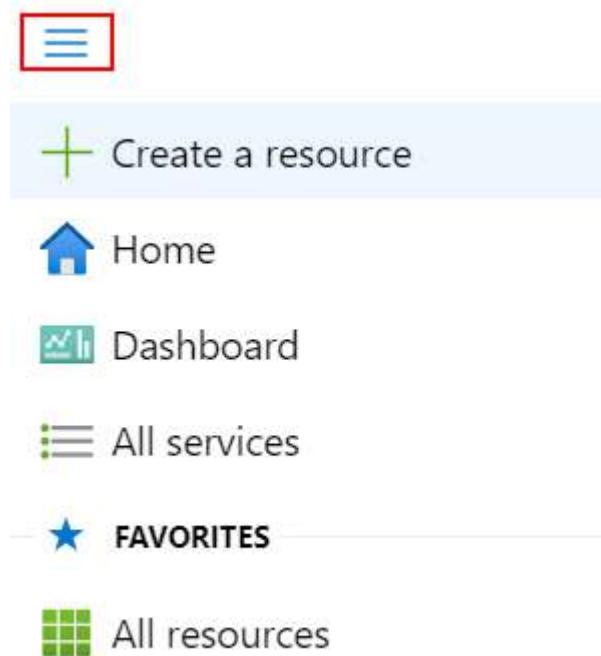
The Azure portal is a web-based, unified console that provides an alternative to command-line tools. With the Azure portal, you can manage your Azure subscription using a graphical user interface. You can build, manage, and monitor everything from simple web apps to complex cloud deployments. Create custom dashboards for an organized view of resources. Configure accessibility options for an optimal experience.

The Azure portal is designed for resiliency and continuous availability. It has a presence in every Azure datacenter. This configuration makes the Azure portal resilient to individual datacenter failures and avoids network slow-downs by being close to users. The Azure portal updates continuously and requires no downtime for maintenance activities.

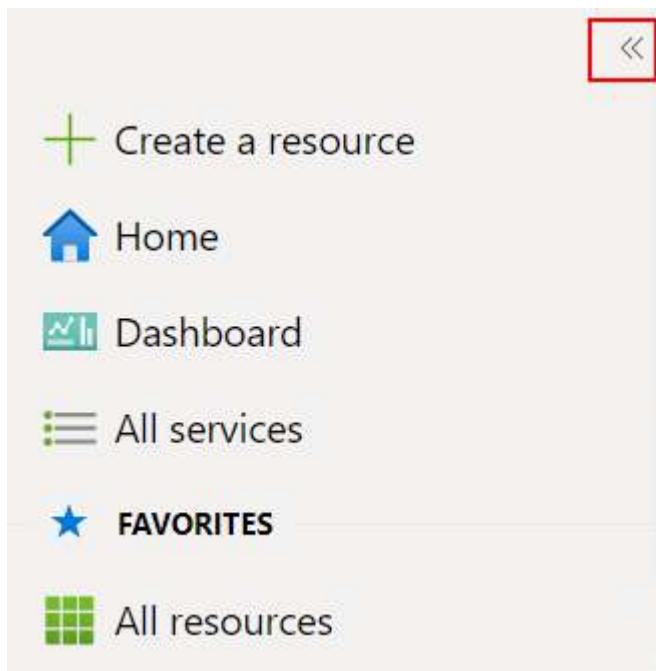
## Azure portal menu

You can choose the default mode for the portal menu. It can be docked or it can act as a flyout panel.

When the portal menu is in flyout mode, it's hidden until you need it. Select the menu icon to open or close the menu.



If you choose docked mode for the portal menu, it will always be visible. You can collapse the menu to provide more working space.



## Azure Home

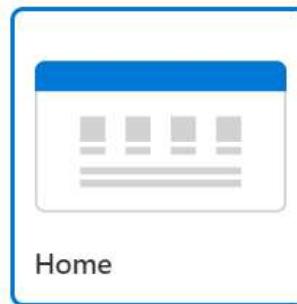
As a new subscriber to Azure services, the first thing you see after you sign in to the portal is Azure Home. This page compiles resources that help you get the most from your Azure subscription. We include links to free online courses, documentation, core services, and useful sites for staying current and managing change for your organization. For quick and easy access to work in progress, we also show a list of your most recently visited resources.

You can't customize the Home page, but you can choose whether to see Home or Dashboard as your default view. The first time you sign in, there's a prompt at the top of the page where you can save your preference. You can change your startup page selection at any time in Portal settings.

### Startup views

Choose your portal landing page and the directory that will load on startup.

Startup page



# Azure Dashboard

Dashboards provide a focused view of the resources in your subscription that matter most to you. We've given you a default dashboard to get you started. You can customize this dashboard to bring the resources you use frequently into a single view. Any changes you make to the default view affect your experience only. However, you can create additional dashboards for your own use, or publish your customized dashboards and share them with other users in your organization.

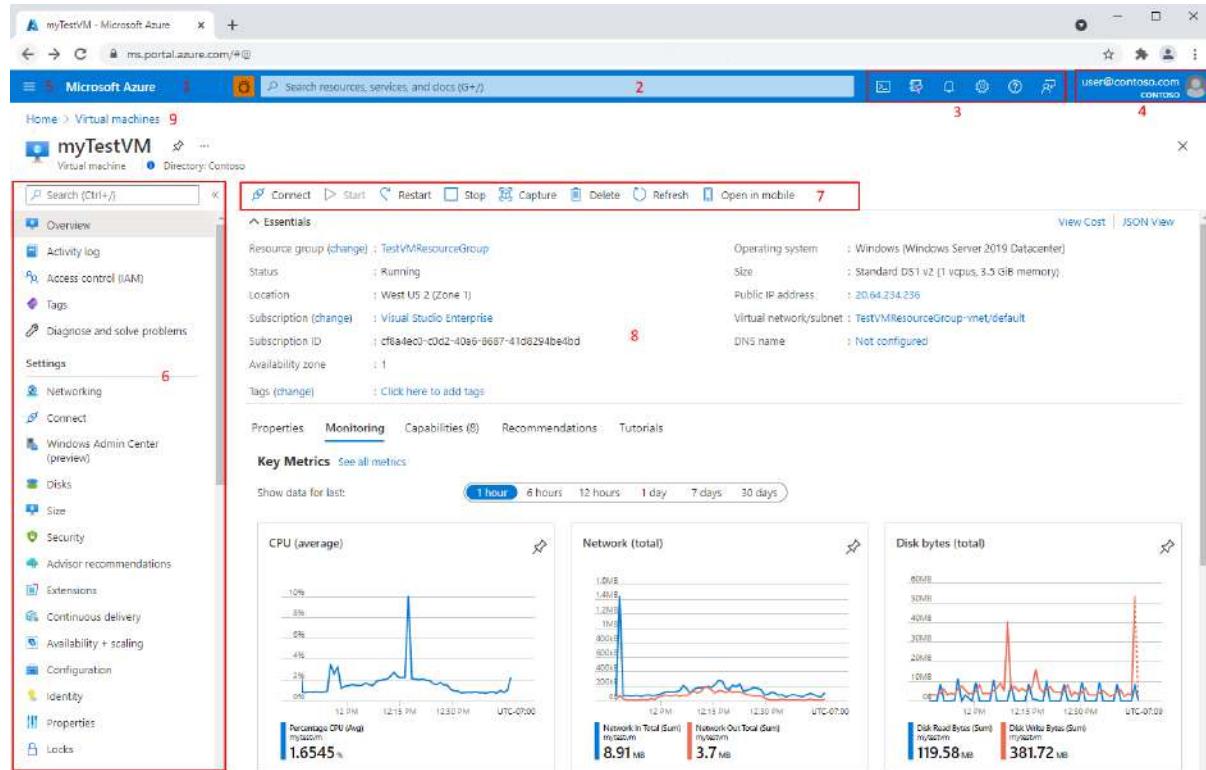
As noted above, you can set your startup page to Dashboard if you want to see your most recently used dashboard when you sign in to the Azure portal.

## Getting around the portal

It's helpful to understand the basic portal layout and how to interact with it. Here, we'll introduce the components of the user interface and some of the terminology we use to give instructions.

The Azure portal menu and page header are global elements that are always present. These persistent features are the "shell" for the user interface associated with each individual service or feature and the header provides access to global controls. The configuration page (sometimes referred to as a "blade") for a resource may also have a resource menu to help you move between features.

The figure below labels the basic elements of the Azure portal, each of which are described in the following table. In this example, the current focus is a virtual machine, but the same elements apply no matter what type of resource or service you're working with.





Create a resource **10**

Home

Dashboard

All services

**FAVORITES** **11**

All resources

Key Description

---

1 Page header. Appears at the top of every portal page and holds global elements.

---

2 Global search. Use the search bar to quickly find a specific resource, a service, or documentation.

---

3 Global controls. Like all global elements, these features persist across the portal and include: Cloud Shell, subscription filter, notifications, portal settings, help and support, and send us feedback.

---

4 Your account. View information about your account, switch directories, sign out, or sign in with a different account.

---

5 Azure portal menu. This global element can help you to navigate between services. Sometimes referred to as the sidebar. (Items 9 and 10 in this list appear in this menu.)

---

- 
- 6 Resource menu. Many services include a resource menu to help you manage the service. You may see this element referred to as the left pane. Here, you'll see commands that are contextual to your current focus.
- 
- 7 Command bar. These controls are contextual to your current focus.
- 
- 8 Working pane. Displays details about the resource that is currently in focus.
- 
- 9 Breadcrumb. You can use the breadcrumb links to move back a level in your workflow.
- 
- 10 Master control to create a new resource in the current subscription. Expand or open the Azure portal menu to find + Create a resource. You can also find this option on the Home page. Then, search or browse the Azure Marketplace for the resource type you want to create.
- 
- 11 Your favorites list in the Azure portal menu.

## Get started with services

If you're a new subscriber, you'll have to create a resource before there's anything to manage. Select + Create a resource to view the services available in the Azure Marketplace. You'll find hundreds of applications and services from many providers here, all certified to run on Azure.

We pre-populate your Favorites in the sidebar with links to commonly used services. To view all available services, select All services from the sidebar.

**Tip:** The quickest way to find a resource, service, or documentation is to use Search in the global header.

# Azure Portal Architecture

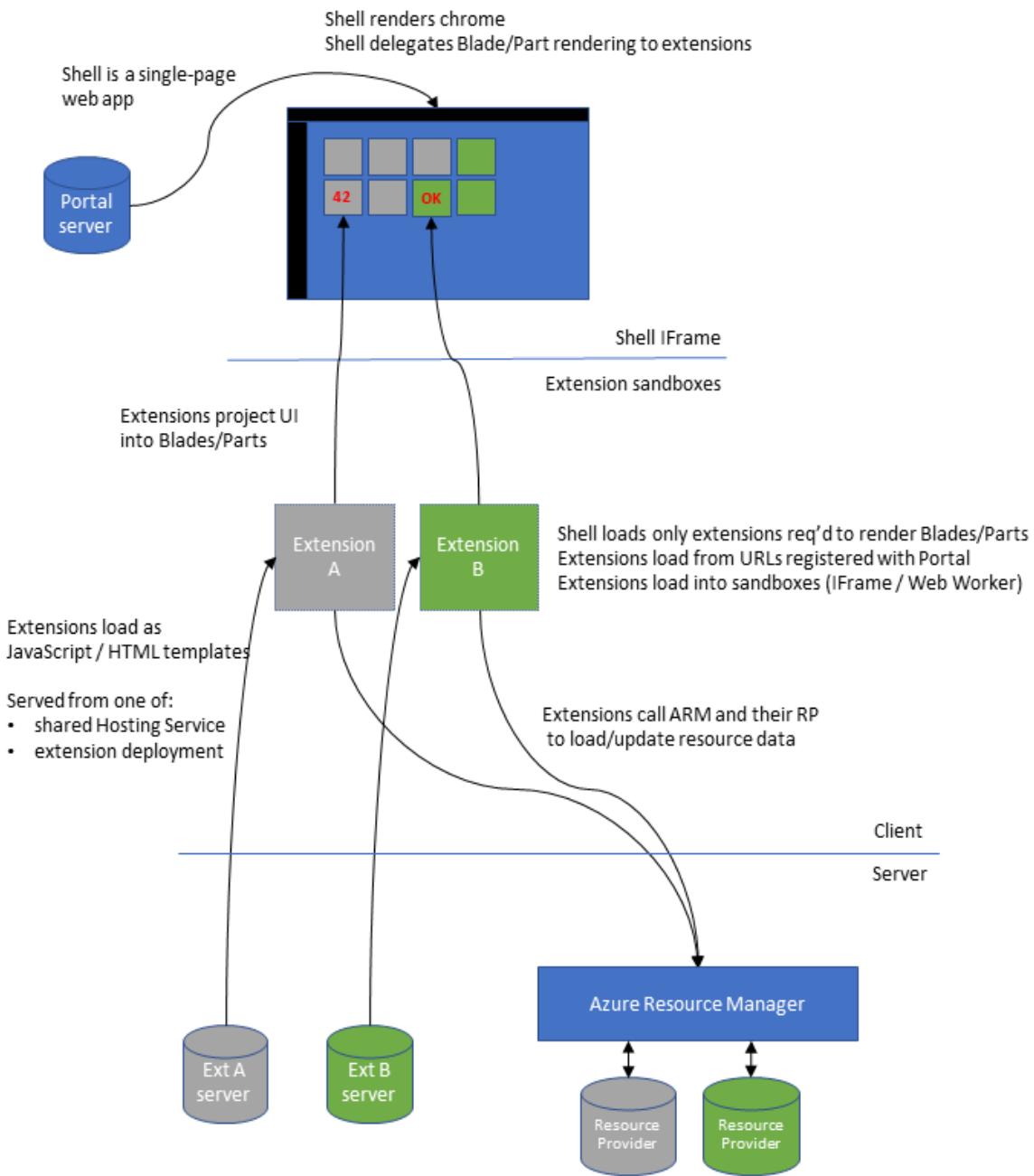
## Azure Portal - A composed web application

The Azure Portal web application is based on a UI composition system whose primary design goal is to enable the integration of UI built by hundreds of teams into a single, robust single-page web application.

With this system, a team develops a UI extension to plug into and extend the UI of the Azure Portal. Teams develop and refine UI iteratively and can choose a deployment cadence that suits their team schedule and their customer needs. They can safely link from their UI to UI's constructed by other teams, resulting in a Portal application that -- to the Azure user -- appears to have been built by a single team. Any bug in a team's UI has only a local impact on that team's UI and does not impact the availability/reliability of the larger Azure Portal UX or that of any other UI extension.

## The Portal Shell

The Azure Portal web application is designed to the single-page application pattern, where UI is generated via client-side-evaluated JavaScript and dynamic HTML. The Azure Portal "Shell" is the client-side JavaScript that controls the overall rendering of UI in the browser. The Shell is responsible for rendering the chrome of the Azure Portal (the navigation menu on the left and bar at the top). Any team- or service-specific UI is developed in UI extensions as Blades (pages or windows) and Parts (tiles). Based on user interaction with the Azure Portal UI, the Shell determines which Blades/Parts are to be displayed and it delegates the rendering of these Blades/Parts to the appropriate extension(s).



## UI extensions

UI extensions are simple, static web applications. Static means that UI extensions do no server-side UI-generation. UI extensions are developed as client-side-evaluated TypeScript that compiles to JavaScript, as HTML templates, and as LESS stylesheets that compile to CSS. When the Shell determines that it needs to display a Blade or Part from UI extension "A", the JIT loads an HTML page that collects the extension JavaScript/HTML/CSS for that extension. The Shell loads the HTML page from an endpoint URL registered centrally as part of the Azure Portal's configuration. The Shell then directs the extension and its client-side JavaScript to render the required Blade or Part. When the user navigates in such

a way that no Blades/Parts from extension "A" are in use, the Shell can unload the UI extension from the browser, allowing the larger Azure Portal app to reclaim browser memory and network connections.

## UI extension isolation

The business logic and UI generation accomplished by UI extensions is isolated from the Azure Portal "chrome" and from the UI of other UI extensions. This is important for three reasons:

- Security - UI extension HTTP access can be restricted to specific origins/endpoints. UI extension JavaScript can be isolated to their own JavaScript heap
- Reliability - UI extension isolation limits the user impact of UI extension bugs
- Scale - Separating UI extension JavaScript allows the UI extension to be "unloaded" from the browser

In 2013, when the Azure Portal was designed, the only browser facility suitable for client-side JavaScript/DOM isolation was the <IFRAME> element. Unfortunately, browsers circa 2013 did not scale performance-wise to the number of Iframes required by the many Parts (tiles) suggested by the 2013 Azure Portal UX design. For this reason, UI extensions are loaded into the browser into non-visible Iframes. Rather than using dynamic HTML techniques that require direct DOM access by the UI extension JavaScript, UI extensions "project" UI using Ibiza FX controls and HTML templates. Such "projected" UI is rendered in the single, visible Iframe that is managed by the Shell.

NOTE: Since browsers circa 2018 scale better Iframes and since the Ibiza UX design is pivoting towards full-screen Blades, the Ibiza team continues to invest in a more conventional use of Iframes, where UI extensions can access the DOM directly and can craft their UI generation following standard web development patterns and OSS libraries.

## Projecting Blade and Part UI

UI extensions develop their Blades and Parts following the MVVM pattern.

- The "view" is defined as a Blade/Part-specific HTML template. The HTML template typically arranges uses of FX controls in the Blade/Part content area.
- The HTML template and FX controls are bound to a UI-extension-developed ViewModel TypeScript class, which is where the UI extension business logic is isolated from the JavaScript of the larger Portal application and from other UI extensions.
- The ViewModel frequently includes "model" data loaded via AJAX from the cloud, though most often it is loaded from the Azure Resource Manager or from the team's/service's Resource Provider.

UI extensions develop a Blade or Part to this pattern by developing a TypeScript class adorned with a TypeScript decorator, as in the following code.

```
@TemplateBlade.Decorator({  
    htmlTemplate: "./WebsiteDetails.html"
```

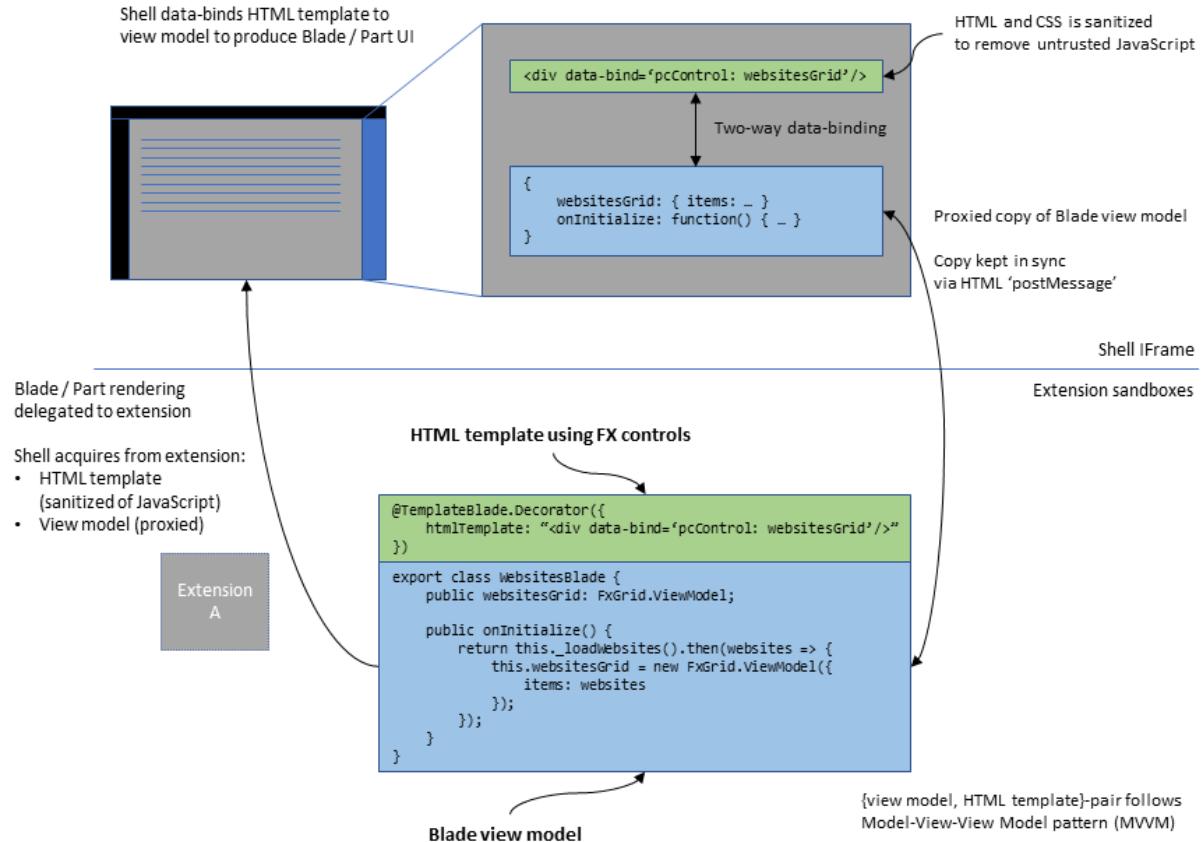
```
})
export class WebsiteDetailsTemplateBlade {
    public title = "Website details";
    public subtitle: string;

    public context: TemplateBlade.Context<void, WebsitesArea.DataContext>;

    public onInitialize() {
        return this._loadWebsiteDetails();
    }
}
```

**NOTE:** Blades and Parts were previously developed by authoring XAML that describes the mapping from a Blade / Part name to its corresponding ViewModel TypeScript class and its associated "view" HTML template. This XAML API (named "PDL" for Portal Definition Language) was found to be developer-unfriendly in that it required that all three artifacts -- the XAML file, the TypeScript class file and the HTML template file -- be managed separately and kept in sync. The new "no-PDL" TypeScript decorator APIs allow for a Blade or Part to be developed in a single TypeScript file.

Now, when a UI extension's Blade or Part is to be displayed, the Shell instantiates in that UI extension's IFrame an instance of the Blade / Part TypeScript class, also known as the ViewModel. To "project" this Blade/Part UI into the Shell-managed visible IFrame that the user sees, the Shell makes use of a simple object-remoting API. Here, the Blade / Part "view" and ViewModel are copied and sent via the HTML postMessage API to the visible IFrame managed by the Shell. It is in the Shell-managed, visible IFrame that the "view" and ViewModel are two-way bound, using the Knockout.js OSS library that is located at <https://knockoutjs.com/>.



Because most UI is dynamic, like Forms that the user updates or like Grids/Lists that are refreshed to reflect new/updated server data, changes to the ViewModel are kept consistent between the Shell and UI extension IFrames. The object-remoting system detects changes to Knockout.js observables that are embedded in the ViewModel, computes diffs between the two ViewModel copies and uses postMessage to send diff-grams between the two ViewModel copies. Beyond the conventional use of the Knockout.js library by the UI extension and its ViewModel class, complexities of the object-remoting system are hidden from the UI extension developer.

## Secure UI per service

The security model for UI extensions builds upon the standard same-origin policy that is supported by all browsers and is the basis for today's web applications. A UI extension's homepage URL is typically located on an origin specific to that UI extension and its resource provider. This HTML page can only issue HTTPS calls to its origin domain and any origins that allow COR calls from the UI extension's origin.

In practice, HTTPS calls from UI extensions are made from the client to load "model" data, and the HTTPS calls are typically directed to the following locations.

- Using CORs, to the Azure Resource Manager (ARM) and/or to the service's Resource Provider (RP);
- Less common, not recommended -- Using same-origin, to HTTP endpoints that are extension controllers that are dedicated to the operation of the UI extension.

In any of these cases, the HTTPS call includes an AAD token that authorizes the UI extension to act on behalf of the user against those Azure resource types that the UI extension supports. The AAD token is obtained during AAD single-sign-on authentication that precedes the loading of the Portal Shell. When a UI extension is loaded into its client-side IFrame and asked to render a Blade or a Part, the UI extension typically calls an FX API with which it can acquire an AAD token that is scoped to that UI extension. To load "model" data, the UI extension then issues HTTP calls carrying this token to ARM, to its RP or to its extension controller.

## Linking and navigating within the Portal

Frequently, user interactions with the Portal chrome and within Blade/Part UI will cause in-Portal navigation to a new Blade. This navigation is accomplished via FX APIs, as in the following example.

```
public onClick() {
    const { container, parameters } = this.context;

    container.openBlade(BladeReferences.forBlade("WebsiteDetailsBlade").createReference({
        parameters: { resourceId: parameters.resourceId },
    }));
}
```

There are two important concepts regarding navigation that are demonstrated here. First, in-Portal navigation is accomplished by using an FX TypeScript API available to UI-extension-authored Blades and Parts instead of by using URL. Second, the API requires the following uses of a code-generated Blade reference types.

- To identify the target Blade in question
- To provide a compiler-verified API for the Blade's parameters

For every Blade and Part developed in a UI extension, Ibiza tooling will code-generate a corresponding “BladeReferenceTypes.d.ts” or “PartReferenceTypes.d.ts” files that can be utilized with FX APIs to open a Blade and to pin a Part respectively, as in the following example.

```
import * as PartPinner from "Fx/Pinner";
import { PartReferences } from "Fx/Composition";

public onPinButtonClick() {
    const { parameters } = this.context;

    PartPinner.pin([
        PartReferences.forPart("WebsitePart").createReference({
            parameters: { resourceId: parameters.resourceId },
        })
    ])
}
```

```
]);  
}
```

These APIs and associated code-generation are critical to integrating UI and UX across Azure services. Similar Blade and Part reference types useful to extension "A" for navigating among its Blades/Parts can be employed by extension "B" to link to Blades from "A". All that is necessary is for extension "A" to redistribute a code package containing the following.

- A PDE file emitted as part of extension "A"'s build
- A TypeScript definition file for those API types used in the construction of extension "A"'s exported Blades and Parts

## Blade and Part API versioning

Each UI-extension-developed Blade and Part includes TypeScript types that describe the set of parameters with which that Blade/Part can be invoked, as in the following example.

```
export interface WebsiteDetailsBladeParameters {  
    resourceId: string;  
}  
  
public onClick() {  
    const { container, parameters } = this.context;  
    const bladeParameters: WebsiteDetailsBladeParameters = {  
        resourceId: parameters.resourceId  
    };  
  
    container.openBlade(BladeReferences.forBlade("WebsiteDetailsBlade").createReference({  
        parameters:  
            bladeParameters  
    }));  
}
```

These form the APIs for the Blades and Parts exported by extension "A" to those teams who wish to link to extension "A" UI. Like any other API that is produced by one team for the consumption of others, these APIs should be updated only in a backwards-compatible manner. The TypeScript implementation of a Blade in extension "A" continues to support all versions of the parameters type ever published/exported to consuming teams. Typically, extensions follow these best practices.

- Never change the name of a Blade or a Part
- Limit their parameters updates to the addition of parameters that are marked in TypeScript as optional
- Never remove parameters from their Parameters type

With this, extensions preserve the flexibility to evolve their sets of Blades and Parts and their APIs, independent of the Portal team and of team's reusing their UI, and to deploy changes at their own cadence, including backward-compatible API changes.

Additionally, the Ibiza SDK contains APIs that allow for the wholesale replacement of one Blade or Part with new equivalent Blades/Parts. It also has APIs that allow for the safe migration of Blades and Part between UI extensions, for example, when responsibilities for specific UIs transfer between teams.

## Common Portal UX Marketplace and Browse

Beyond Blades and Parts, UI extensions can benefit from other UI integration with the Azure Portal. First, a UI extension team can develop "Marketplace packages". Each of these packages describes an entry that will be displayed in the Azure Portal "Marketplace" UI. A package includes the following.

- Metadata that specifies the UI for the item in the marketplace (icons, text, etc)
- Metadata that locates the Blade that will be invoked once that item is selected by the user in the Marketplace. This Blade is a stylized Form that includes logic that knows how to provision Azure resources
- ARM templates that describe how a resource will be provisioned via ARM.

A UI extension team publishes their N>0 packages to a service, where all such packages/items are available in queryable form to the Azure Portal and to the Marketplace UI. The Marketplace UI is itself implemented as Blades in the "Azure Marketplace" UI extension.

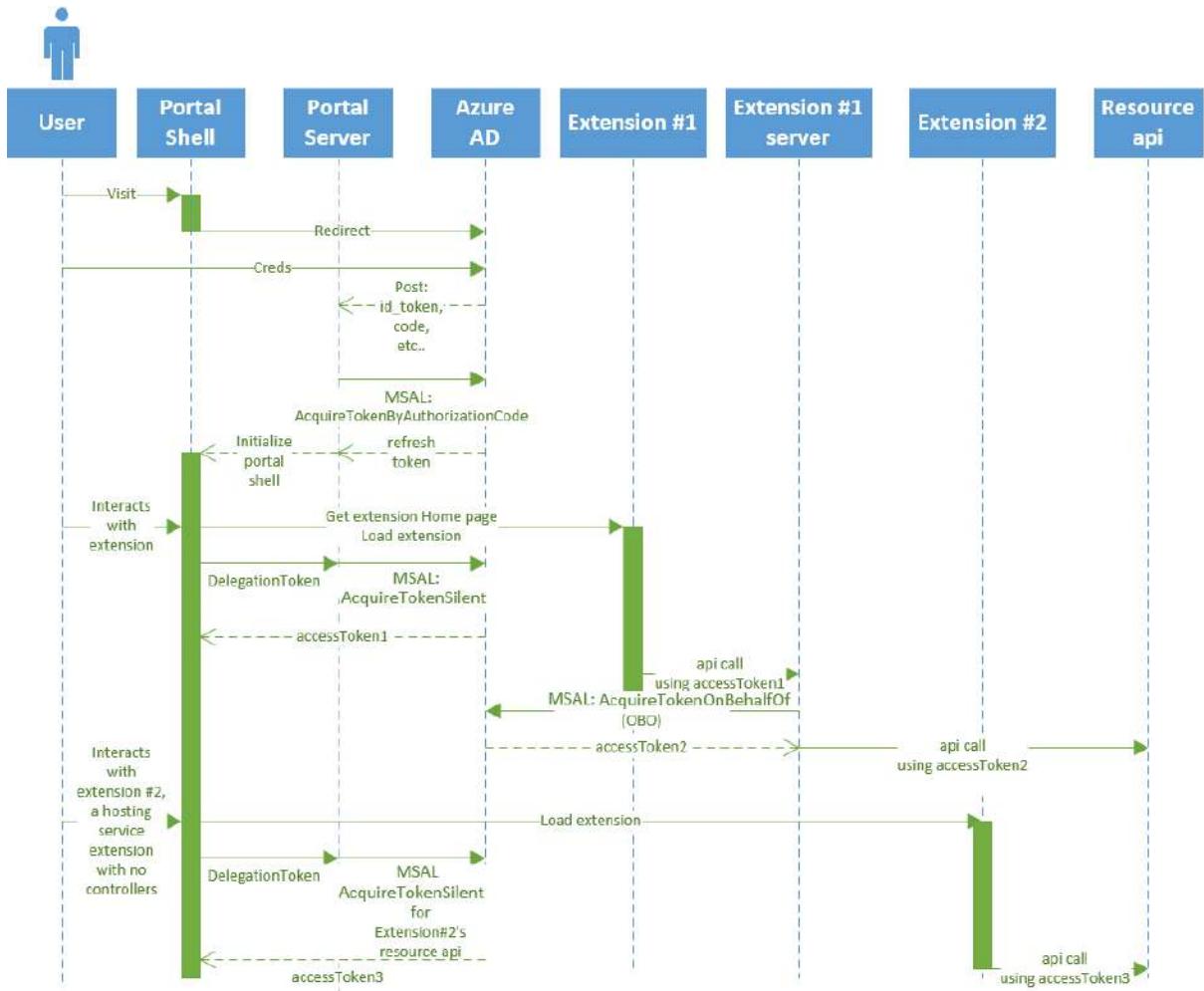
Second, for a given UI extension / ARM resource type to be represented in the standard Azure Portal "Browse" UI, UI extensions develop UI metadata for the resource types that their UI extension supports. This is accomplished with the PDL <AssetType> tag. No-PDL variants of <AssetType> are currently in development. An "asset type" is expressed as metadata so that standard Azure Portal UI, like "Browse" Blades and Parts, can display resources without the need to involve UI extension business logic, which is potentially a performance problem when spanning N resource types.

## Portal Authentication Architecture

### Portal Sign-in flow

The Azure Portal uses Azure AD for authentication. Once the portal obtains a token from AD the user can then use that token to access the various backend services that power the UI. For example, the token can call Azure Resource Manager and the Active Directory Graph.

The following diagram shows the sequence of the authentication events during the sign-in process. It also shows how the portal interacts with the various backend services.



The following is the signin process that is performed by the Portal.

1. User browses to the Portal.
2. The Portal redirects to AAD to sign in.
3. AAD redirects to the portal.azure.com/signin/index/ with an id\_token. The audience of the id\_token is the Azure Portal app.
4. The Portal server exchanges this code for an access token. This refresh token is returned to the browser, as part of the Portal Shell UI and is stored in memory.
5. When the user triggers the loading of the extension, the extension home page is downloaded from the location in the extension configuration, as specified in portalfx-extensions-configuration-overview.md. This provides information required by the Shell to bootstrap the extension UI.
6. When an extension asks for a token, it makes a request to the Portal DelegationToken controller endpoint. This endpoint requires the refresh token that was just acquired, in addition to the extension name and the resource name that the extension requested. This endpoint returns access tokens to the browser.

Sample request:

```
"extensionName":"Microsoft_AAD_IAM",
"resourceName":"self",
"tenant":"9e4917cd-bd32-4371-b1c8-82b5d610f2e2",
"portalAuthorization":"MIIF...."
```

Sample response:

```
"value":{  
    "authHeader":"Bearer eyJ0...",  
    "authorizationHeader":"Bearer ...",  
    "expiresInMs":3299000,  
    "refreshToken":"MIIF...",  
    "error":null,  
    "errorMessage":null  
},  
"portalAuthorization":"MIIF..."
```

NOTE: Tokens are cached in memory on the server. In this example, the resourceName: self indicates that this extension only calls itself from the client.

7. Now the extension's client side can call server side API's, or call external services directly.

- Server side API

The extension's server-side code can exchange its current access token for another access token that allows it to use resources.

- Direct external services call

The PortalFx's client side ajax wrapper makes the DelegationToken call to get a token for the specified resource. In this case, the PortalFx team creates the app registration for the extension and manages permissions for the API's.

The framework keeps track of token expiration. When the user interacts with the site in a way that results in an API call, and the token is about to expire, the framework makes the call to DelegationToken again to get new access tokens. The extension uses the PortalFx's client side ajax wrapper.

The Portal signs the user into the last-used directory, the home directory for AAD accounts, or the first directory it gets from ARM for MSA accounts. It also loads the Startboard and all extensions.

Navigating to a new extension repeats this process, beginning at triggering the loading of the extension. If the user revisits an extension, a client side in-memory token cache is used instead of making another request to the Portal DelegationToken controller endpoint. This cache is lost on page refresh.

NOTE: Cookies are not used for authentication.

## Azure Cloud Shell

Azure Cloud Shell is an interactive, authenticated, browser-accessible shell for managing Azure resources. It provides the flexibility of choosing the shell experience that best suits the way you work, either Bash or PowerShell.

You can access the Cloud Shell in three ways:

- Direct link: Open a browser to <https://shell.azure.com>.
- Azure portal: Select the Cloud Shell icon on the Azure portal:



- Code snippets: On [docs.microsoft.com](https://docs.microsoft.com) and Microsoft Learn, select the Try It button that appears with Azure CLI and Azure PowerShell code snippets:

[az account show](#)

## [Get-AzSubscription](#)

The Try It button opens the Cloud Shell directly alongside the documentation using Bash (for Azure CLI snippets) or PowerShell (for Azure PowerShell snippets).

To run the command, use Copy in the code snippet, use Ctrl+Shift+V (Windows/Linux) or Cmd+Shift+V (macOS) to paste the command, and then press Enter.

## Features

### Browser-based shell experience

Cloud Shell enables access to a browser-based command-line experience built with Azure management tasks in mind. Leverage Cloud Shell to work untethered from a local machine in a way only the cloud can provide.

### Choice of preferred shell experience

Users can choose between Bash or PowerShell.

1. Select Cloud Shell.

The screenshot shows the Microsoft Azure portal interface. At the top, there is a navigation bar with icons for back, forward, home, and search, followed by the URL 'https://portal.azure.com/'. A red box highlights the 'Cloud Shell' icon, which is a downward-pointing arrow. Below the navigation bar is a blue header bar with the text 'Microsoft Azure' and a search bar that says 'Search resources, services, and docs (G+)'. The main content area is titled 'Azure services' and contains several service icons: 'Create a resource' (plus sign), 'Azure Database for MySQL' (database icon), 'Azure Active Directory' (person icon), 'Digital Twins' (grid icon), 'App Services' (globe icon), 'Recent' (clock icon), and 'Resource groups' (cube icon). Below these are 'Service Health' (heart icon) and 'Quickstart Center' (rocket icon). There is also a link 'More services' with a right-pointing arrow. Underneath this is a section titled 'Recent resources' with a table:

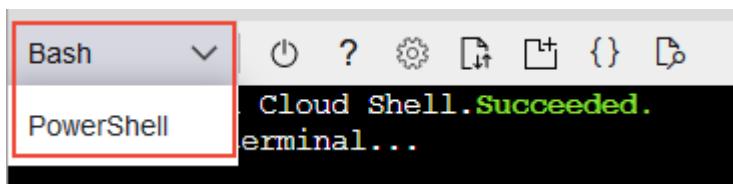
NAME	TYPE	LAST VIEWED
cs4316e81020662x41cbxb95	Storage account	2 d ago

At the bottom, there is a 'Navigate' section with links for 'Subscriptions' (key icon), 'Resource groups' (cube icon), and 'All resources' (grid icon).

2. Select Bash or PowerShell.

The screenshot shows the 'Welcome to Azure Cloud Shell' screen. It features a large downward-pointing arrow icon. Below it is the text 'Welcome to Azure Cloud Shell'. A message reads: 'Select Bash or PowerShell. You can change shells any time via the environment selector in the Cloud Shell toolbar. The most recently used environment will be the default for your next session.' At the bottom, there are two buttons: 'Bash' and 'PowerShell', both of which are highlighted with a red box.

After first launch, you can use the shell type drop-down control to switch between Bash and PowerShell:



## Authenticated and configured Azure workstation

Cloud Shell is managed by Microsoft so it comes with popular command-line tools and language support. Cloud Shell also securely authenticates automatically for instant access to your resources through the Azure CLI or Azure PowerShell cmdlets.

## Integrated Cloud Shell editor

Cloud Shell offers an integrated graphical text editor based on the open-source Monaco Editor. Simply create and edit configuration files by running code . for seamless deployment through Azure CLI or Azure PowerShell.

## Multiple access points

Cloud Shell is a flexible tool that can be used from:

- portal.azure.com
- shell.azure.com
- Azure CLI documentation
- Azure PowerShell documentation
- Azure mobile app
- Visual Studio Code Azure Account extension

## Connect your Microsoft Azure Files storage

Cloud Shell machines are temporary, but your files are persisted in two ways: through a disk image, and through a mounted file share named clouddrive. On first launch, Cloud Shell prompts to create a resource group, storage account, and Azure Files share on your behalf. This is a one-time step and will be automatically attached for all sessions. A single file share can be mapped and will be used by both Bash and PowerShell in Cloud Shell.

**Note:** Azure storage firewall is not supported for cloud shell storage accounts.

## Concepts

- Cloud Shell runs on a temporary host provided on a per-session, per-user basis
- Cloud Shell times out after 20 minutes without interactive activity
- Cloud Shell requires an Azure file share to be mounted
- Cloud Shell uses the same Azure file share for both Bash and PowerShell
- Cloud Shell is assigned one machine per user account
- Cloud Shell persists \$HOME using a 5-GB image held in your file share
- Permissions are set as a regular Linux user in Bash

## Compliance

### Encryption at rest

All Cloud Shell infrastructure is compliant with double encryption at rest by default. No action is required by users.

## Pricing

The machine hosting Cloud Shell is free, with a pre-requisite of a mounted Azure Files share. Regular storage costs apply.

## Azure security baseline for Cloud Shell

This security baseline applies guidance from the [Azure Security Benchmark version 2.0](#) to Cloud Shell. The Azure Security Benchmark provides recommendations on how you can secure your cloud solutions on Azure. The content is grouped by the security controls defined by the Azure Security Benchmark and the related guidance applicable to Cloud Shell.

**Note:** Controls not applicable to Cloud Shell, and those for which the global guidance is recommended verbatim, have been excluded.

## Network Security

### NS-1: Implement security for internal traffic

**Guidance:** When you deploy Cloud Shell resources, create or use an existing virtual network. Ensure that all Azure virtual networks follow an enterprise segmentation principle that aligns with the business risks. Isolate any system that might incur higher risk for the organization within its own virtual network. Secure the virtual network sufficiently with a network security group (NSG) and/or Azure Firewall.

Use Microsoft Defender for Cloud adaptive network hardening to recommend NSG configurations that limit ports and source IPs based on external network traffic.

**Responsibility:** Customer

Microsoft Defender for Cloud monitoring: None

### NS-4: Protect applications and services from external network attacks

**Guidance:** Protect your Cloud Shell resources against attacks from external networks. External attacks can include distributed denial of service (DDoS) attacks, application-specific attacks, and unsolicited and potentially malicious internet traffic.

Use Azure Firewall to protect applications and services against potentially malicious traffic from the internet and other external locations.

Protect your assets against DDoS attacks by enabling DDoS Protection Standard on your Azure virtual networks. Use Microsoft Defender for Cloud to detect misconfiguration risks to your network-related resources.

Cloud Shell in an Azure virtual network is an optional experience and isn't set up by default. Cloud Shell doesn't run web applications. You don't have to configure any settings or deploy any network services to protect from external network attacks that target web applications.

**Responsibility:** Customer

Microsoft Defender for Cloud monitoring: None

## Identity Management

### IM-1: Standardize Azure Active Directory as the central identity and authentication system

Guidance: Cloud Shell uses Azure Active Directory (Azure AD) as its default identity and access management service. Standardize Azure AD to govern your organization's identity and access management in:

- Microsoft Cloud resources. Resources include:
  - The Azure portal
  - Azure Storage
  - Azure Linux and Windows virtual machines
  - Azure Key Vault
  - Platform-as-a-service (PaaS)
  - Software-as-a-service (SaaS) applications
- Your organization's resources, such as applications on Azure or your corporate network resources.

Securing Azure AD should be a high priority for your organization's cloud security practice. Azure AD provides an identity secure score to help you compare your identity security posture to Microsoft's best practice recommendations. Use the score to gauge how closely your configuration matches best practice recommendations, and to make improvements in your security posture.

Note: Azure AD supports external identities that allow users without Microsoft accounts to sign in to their applications and resources.

Responsibility: Customer

Microsoft Defender for Cloud monitoring: None

### IM-7: Eliminate unintended credential exposure

Guidance: Cloud Shell lets customers run code, deploy configurations, or persist data that potentially contain identities or secrets. Use Credential Scanner to identify these credentials. Credential Scanner encourages moving discovered credentials to secure locations like Azure Key Vault.

For GitHub, you can use the native secret scanning feature to identify credentials or other secrets in code.

Responsibility: Customer

Microsoft Defender for Cloud monitoring: None

## Privileged Access

### PA-6: Use privileged access workstations

Guidance: Secured, isolated workstations are critical for security of sensitive roles like administrator, developer, and critical service operator. Use highly secured user workstations and Azure Bastion for administrative tasks.

Use Azure AD, Microsoft Defender Advanced Threat Protection (ATP), or Microsoft Intune to deploy a secure and managed user workstation for administrative tasks. You can manage secured workstations centrally to enforce a security configuration that includes:

- Strong authentication
- Software and hardware baselines
- Restricted logical and network access

Responsibility: Customer

Microsoft Defender for Cloud monitoring: None

## Data Protection

### DP-3: Monitor for unauthorized transfer of sensitive data

Guidance: Not applicable. Cloud Shell supports transferring customer data, but doesn't natively support monitoring for unauthorized transfer of sensitive data.

Responsibility: Customer

Microsoft Defender for Cloud monitoring: None

## Asset Management

### AM-1: Ensure the security team has visibility into risks for assets

Guidance: Make sure to grant security teams Security Reader permissions in your Azure tenant and subscriptions, so they can monitor for security risks by using Microsoft Defender for Cloud.

Monitoring for security risks could be the responsibility of a central security team or a local team, depending on how you structure responsibilities. Always aggregate security insights and risks centrally within an organization.

You can apply Security Reader permissions broadly to an entire tenant's Root Management Group, or scope permissions to specific management groups or subscriptions.

Note: Visibility into workloads and services might require more permissions.

Responsibility: Customer

Microsoft Defender for Cloud monitoring: None

**AM-2: Ensure the security team has access to asset inventory and metadata**

**Guidance:** Metadata tags logically organize resources in a taxonomy. Cloud Shell doesn't use tags or let customers apply or use tags.

Cloud Shell doesn't support Azure Resource Manager-based resource deployments or discovery with Azure Resource Graph.

You can use Microsoft Defender for Cloud adaptive application controls to specify which file types a rule applies to.

**Responsibility:** Customer

Microsoft Defender for Cloud monitoring: None

**AM-6: Use only approved applications in compute resources**

**Guidance:** Use Azure Virtual Machine Inventory to automate collecting information about software on virtual machines (VMs). You can get Software Name, Version, Publisher, and Refresh Time from the Azure portal. To access install dates and other information, enable guest-level diagnostics and import the Windows Event Logs into a Log Analytics workspace.

**Responsibility:** Customer

Microsoft Defender for Cloud monitoring: None

## **Logging and Threat Detection**

**LT-3: Enable logging for Azure network activities**

**Guidance:** Even though you can deploy Cloud Shell resources into a virtual network, you can't enforce network traffic by, or pass traffic through, a network security group. You have to disable network policies on the subnet for Cloud Shell to work correctly. For this reason, you can't configure network security group flow logging for Cloud Shell.

Cloud Shell doesn't produce or process Domain Name Service (DNS) query logs.

**Responsibility:** Customer

Microsoft Defender for Cloud monitoring: None

## **Posture and Vulnerability Management**

**PV-6: Perform software vulnerability assessments**

**Guidance:** Cloud Shell can use a third-party solution to do vulnerability assessments on network devices and web applications. Don't use a single, perpetual administrative account when conducting remote scans. Consider implementing just-in-time (JIT) provisioning methodology for the scan account. Protect and monitor credentials for the scan account, and use them only for vulnerability scanning.

As required, export scan results at consistent intervals. Compare the results with previous scans to verify that vulnerabilities are remediated.

Responsibility: Customer

Microsoft Defender for Cloud monitoring: None

#### PV-7: Rapidly and automatically remediate software vulnerabilities

Guidance: For third-party software, use a third-party patch management solution or System Center Updates Publisher for Configuration Manager.

For a full list of features and tools, see: [Cloud Shell features:/azure/cloud-shell/features](#)

Responsibility: Customer

Microsoft Defender for Cloud monitoring: None

#### PV-8: Conduct regular attack simulation

Guidance: Conduct penetration testing or red team activities on your Azure resources as needed, and ensure remediation of all critical security findings.

Follow the Microsoft Cloud Penetration Testing Rules of Engagement to ensure your penetration tests don't violate Microsoft policies. Use Microsoft's Red Teaming strategy and execution. Do live site penetration testing against Microsoft-managed cloud infrastructure, services, and applications.

Responsibility: Customer

Microsoft Defender for Cloud monitoring: None

## Cost Management

### What is Cost Management + Billing?

By using the Microsoft cloud, you can significantly improve the technical performance of your business workloads. It can also reduce your costs and the overhead required to manage organizational assets. However, the business opportunity creates a risk because of the potential for waste and inefficiencies that are introduced into your cloud deployments. Cost Management + Billing is a suite of tools provided by Microsoft that help you analyze, manage, and optimize the costs of your workloads. Using the suite helps ensure that your organization is taking advantage of the benefits provided by the cloud.

You can think of your Azure workloads like the lights in your home. When you leave to go out for the day, are you leaving the lights on? Could you use different bulbs that are more efficient to help reduce your monthly energy bill? Do you have more lights in one room than are needed? You can use Cost Management + Billing to apply a similar thought process to the workloads used by your organization.

With Azure products and services, you only pay for what you use. As you create and use Azure resources, you're charged for the resources. Because of the deployment ease for new

resources, the costs of your workloads can jump significantly without proper analysis and monitoring. You use Cost Management + Billing features to:

- Conduct billing administrative tasks such as paying your bill
- Manage billing access to costs
- Download cost and usage data that was used to generate your monthly invoice
- Proactively apply data analysis to your costs
- Set spending thresholds
- Identify opportunities for workload changes that can optimize your spending



## Understand Azure Billing

Azure Billing features are used to review your invoiced costs and manage access to billing information. In larger organizations, procurement and finance teams usually conduct billing tasks.

A billing account is created when you sign up to use Azure. You use your billing account to manage your invoices, payments, and track costs. You can have access to multiple billing accounts. For example, you might have signed up for Azure for your personal projects. So, you might have an individual Azure subscription with a billing account. You could also have access through your organization's Enterprise Agreement or Microsoft Customer Agreement. For each scenario, you would have a separate billing account.

## Billing accounts

The Azure portal currently supports the following types of billing accounts:

- Microsoft Online Services Program: An individual billing account for a Microsoft Online Services Program is created when you sign up for Azure through the Azure website. For example, when you sign up for an Azure Free Account, account with pay-as-you-go rates or as a Visual studio subscriber.
- Enterprise Agreement: A billing account for an Enterprise Agreement is created when your organization signs an Enterprise Agreement (EA) to use Azure.
- Microsoft Customer Agreement: A billing account for a Microsoft Customer Agreement is created when your organization works with a Microsoft representative to sign a Microsoft Customer Agreement. Some customers in select regions, who sign up through the Azure website for an account with pay-as-you-go rates or upgrade their Azure Free Account may have a billing account for a Microsoft Customer Agreement as well.

## Understand Cost Management

Although related, billing differs from cost management. Billing is the process of invoicing customers for goods or services and managing the commercial relationship.

Cost Management shows organizational cost and usage patterns with advanced analytics. Reports in Cost Management show the usage-based costs consumed by Azure services and third-party Marketplace offerings. Costs are based on negotiated prices and factor in reservation and Azure Hybrid Benefit discounts. Collectively, the reports show your internal and external costs for usage and Azure Marketplace charges. Other charges, such as reservation purchases, support, and taxes aren't yet shown in reports. The reports help you understand your spending and resource use and can help find spending anomalies. Predictive analytics are also available. Cost Management uses Azure management groups, budgets, and recommendations to show clearly how your expenses are organized and how you might reduce costs.

You can use the Azure portal or various APIs for export automation to integrate cost data with external systems and processes. Automated billing data export and scheduled reports are also available.

## Plan and control expenses

The ways that Cost Management help you plan for and control your costs include: Cost analysis, budgets, recommendations, and exporting cost management data.

You use cost analysis to explore and analyze your organizational costs. You can view aggregated costs by organization to understand where costs are accrued and to identify spending trends. And you can see accumulated costs over time to estimate monthly, quarterly, or even yearly cost trends against a budget.

Budgets help you plan for and meet financial accountability in your organization. They help prevent cost thresholds or limits from being surpassed. Budgets can also help you inform others about their spending to proactively manage costs. And with them, you can see how spending progresses over time.

Recommendations show how you can optimize and improve efficiency by identifying idle and underutilized resources. Or, they can show less expensive resource options. When you act on the recommendations, you change the way you use your resources to save money. To act, you first view cost optimization recommendations to view potential usage inefficiencies. Next, you act on a recommendation to modify your Azure resource use to a more cost-effective option. Then you verify the action to make sure that the change you make is successful.

If you use external systems to access or review cost management data, you can easily export the data from Azure. And you can set a daily scheduled export in CSV format and store the data files in Azure storage. Then, you can access the data from your external system.

## Additional Azure tools

Azure has other tools that aren't a part of the Cost Management + Billing feature set. However, they play an important role in the cost management process.

- Azure Pricing Calculator - Use this tool to estimate your up-front cloud costs.
- Azure Migrate - Assess your current datacenter workload for insights about what's needed from an Azure replacement solution.
- Azure Advisor - Identify unused VMs and receive recommendations about Azure reserved instance purchases.
- Azure Hybrid Benefit - Use your current on-premises Windows Server or SQL Server licenses for VMs in Azure to save.

## How to optimize your cloud investment with Cost Management

Cost Management gives you the tools to plan for, analyze and reduce your spending to maximize your cloud investment. This document provides you with a methodical approach to cost management and highlights the tools available to you as you address your organization's cost challenges. Azure makes it easy to build and deploy cloud solutions. However, it's important that those solutions are optimized to minimize the cost to your organization. Following the principles outlined in this document and using our tools will help to make sure your organization is prepared for success.

## Methodology

Cost management is an organizational problem and should be an ongoing practice that begins before you spend money on cloud resources. To successfully implement cost management and optimize costs, your organization must:

- Be prepared with the proper tools for success
- Be accountable for costs
- Take appropriate action to optimize spending

Three key groups, outlined below, must be aligned in your organization to make sure that you successfully manage costs.

- Finance - People responsible for approving budget requests across the organization based on cloud spending forecasts. They pay the corresponding bill and assign costs to various teams to drive accountability.
- Managers - Business decision makers in an organization that need to understand cloud spending to find the best spending results.
- App teams - Engineers managing cloud resources on a day-to-day basis, developing services to meet the organization's needs. These teams need the flexibility to deliver the most value in their defined budgets.

## Key principles

Use the principles outlined below to position your organization for success in cloud cost management.

## Planning

Comprehensive, up-front planning allows you to tailor cloud usage to your specific business requirements. Ask yourself:

- What business problem am I solving?
- What usage patterns do I expect from my resources?

Your answers will help you select the offerings that are right for you. They determine the infrastructure to use and how it's used to maximize your Azure efficiency.

## Visibility

When structured well, Cost Management helps you to inform people about the Azure costs they're responsible for or for the money they spend. Azure has services designed to give you insight into where your money is spent. Take advantage of these tools. They can help you find resources that are underused, remove waste, and maximize cost-saving opportunities.

## Accountability

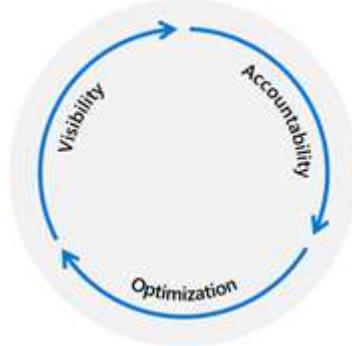
Attribute costs in your organization to make sure that people responsible are accountable for their team's spending. To fully understand your organization's Azure spending, you should organize your resources to maximize insight into cost attribution. Good organization helps to manage and reduce costs and hold people accountable for efficient spending in your organization.

## Optimization

Act to reduce your spending. Make the most of it based on the findings gathered through planning and increasing cost visibility. You might consider purchase and licensing optimizations along with infrastructure deployment changes that are discussed in detail later in this document.

## Iteration

Everyone in your organization must engage in the cost management lifecycle. They need to stay involved on an ongoing basis to optimize costs. Be rigorous about this iterative process and make it a key tenet of responsible cloud governance in your organization.



## Plan with cost in mind

Before you deploy cloud resources, assess the following items:

- The Azure offer that best meets your needs
- The resources you plan to use
- How much they might cost

Azure provides tools to assist you in the assessment process. The tools can give you a good idea of the investment required to enable your workloads. Then you can select the best configuration for your situation.

## Azure onboarding options

The first step in maximizing your experience within Cost Management is to investigate and decide which Azure offer is best for you. Think about how you plan to use Azure in the future. Also consider how you want your billing model configured. Consider the following questions when making your decision:

- How long do I plan to use Azure? Am I testing, or do I plan to build longer-term infrastructure?
- How do I want to pay for Azure? Should I prepay for a reduced price or get invoiced at the end of the month?

## Free

- 12 months of popular free services
- \$200 credit in your billing currency to explore services for 30 days
- 25+ services are always free

## Pay as you go

- No minimums or commitments
- Competitive Pricing
- Pay only for what you use
- Cancel anytime

## Enterprise Agreement

- Options for up-front Azure Prepayment (previously called monetary commitment)
- Access to reduced Azure pricing

## Azure in CSP

- CSP partners are the first point of contact for their customers' needs and the center of the customer relationship
- CSP partners provision new customers, order subscriptions, manage subscriptions, and perform admin tasks on behalf of their customers
- CSP partners bundle services with unique solutions or resell Azure while controlling the pricing, terms and billing

## **Estimate the cost of your solution**

Before you deploy any infrastructure, assess how much your solution will cost. The assessment will help you create a budget for your organization for the workload, up-front. Then you can use a budget over time to benchmark the validity of your initial estimation. And you can compare it with the actual cost of your deployed solution.

## Azure pricing calculator

The Azure pricing calculator allows you to mix and match different combinations of Azure services to see an estimate of the costs. You can implement your solution using different ways in Azure - each might influence your overall spending. Thinking early about all of the infrastructure needs of your cloud deployment helps you use the tool most effectively. It can help you get a solid estimate of your estimated spending in Azure.

## Azure Migrate

Azure Migrate is a service that assesses your organization's current workloads in on-premises datacenters. It gives you insight into what you might need from an Azure replacement solution. First, Migrate analyzes your on-premises machines to determine whether migration is feasible. Then, it recommends VM sizing in Azure to maximize performance. Finally, it also creates a cost estimate for an Azure-based solution.

## Analyze and manage your costs

Keep informed about how your organization's costs evolve over time. Use the following techniques to properly understand and manage your spending.

### Organize resources to maximize cost insights and accountability

A well-planned organizational structure for your Azure billing and resource hierarchies helps to give you a good understanding and control over costs as you create your cloud infrastructure. As you evaluate and create a hierarchy that meets your needs, ask yourself the following questions.

*Which billing hierarchy is available to me and what are the different scopes that I can use?*

Identify the billing arrangement for your organization by determining your Azure offer type.

*If I have multiple teams, how should I organize my subscriptions and resource groups?*

Creating a subscription or resource group for each team is a common practice. They can help you to differentiate costs and hold teams accountable. However, costs are bound to the subscription or resource group.

If you already have teams with multiple subscriptions, consider grouping the subscriptions into management groups to analyze the costs together. Management groups, subscriptions, and resource groups are all part of the Azure RBAC hierarchy. Use them collectively for access control in your teams.

Resources can span across multiple scopes, especially when they're shared by multiple teams or workloads. Consider identifying resources with tags. Tags are discussed further in the next section.

*Do I have Development and Production environments?*

Consider creating Dev/Test subscriptions for your development environments to take advantage of reduced pricing. If the workloads span multiple teams or Azure scopes, consider using tags to identify them.

## Tag shared resources

Tags are a effective way to understand costs that span across multiple teams and Azure scopes. For example, you might have a resource like an email server that many teams use. You can put a shared resource, like the email server, in a subscription that's dedicated to shared resources or put it in an existing subscription. If you put it in an existing subscription, the subscription owner might not want its cost accruing to their team every month. For this example, you can use a tag to identify the resource as being shared.

Similarly, you might also have web apps or environments, such as Test or Production, that use resources across multiple subscriptions owned by different teams. To better understand the full cost of the workloads, tag the resources that they use. When tags are applied properly, you can apply them as a filter in cost analysis to better understand trends. After you plan for resource tagging, you can configure an Azure Policy definition to enforce tagging on resources.

## Use cost analysis

Cost analysis allows you to analyze your organizational costs in-depth by slicing and dicing your costs using standard resource properties. Consider the following common questions as a guide for your analysis. Answering these questions on a regular basis will help you stay more informed and enable more cost-conscious decisions.

- Estimated costs for the current month – How much have I incurred so far this month? Will I stay under my budget?
- Investigate anomalies – Do routine checks to make sure that costs stay within a reasonable range of normal usage. What are the trends? Are there any outliers?
- Invoice reconciliation - Is my latest invoiced cost more than the previous month? How did spending habits change month-over-month?
- Internal chargeback - Now that I know how much I'm being charged, how should those charges be broken down for my organization?

## Export billing data on a schedule

Do you need to import your billing data into an external system, like a dashboard or financial system? Set up automated exports to Azure Storage and avoid manually downloading files every month. You can then easily set up automatic integrations with other systems to keep your billing data in sync.

## Create budgets

After you've identified and analyzed your spending patterns, it's important to begin setting limits for yourself and your teams. Azure budgets give you the ability to set either a cost or usage-based budget with many thresholds and alerts. Make sure to review the budgets that you create regularly to see your budget burn-down progress and make changes as

needed. Azure budgets also allow you to configure an automation trigger when a given budget threshold is reached. For example, you can configure your service to shut down VMs. Or you can move your infrastructure to a different pricing tier in response to a budget trigger.

## Act to optimize

Use the following ways to optimize spending.

### Cut out waste

After you've deployed your infrastructure in Azure, it's important to make sure it is being used. The easiest way to start saving immediately is to review your resources and remove any that aren't being used. From there, you should determine if your resources are being used as efficiently as possible.

### Azure Advisor

Azure Advisor is a service that, among other things, identifies virtual machines with low utilization from a CPU or network usage standpoint. From there, you can decide to either shut down or resize the machine based on the estimated cost to continue running the machines. Advisor also provides recommendations for reserved instance purchases. The recommendations are based on your last 30 days of virtual machine usage. When acted on, the recommendations can help you reduce your spending.

### Size your VMs properly

VM sizing has a significant impact on your overall Azure cost. The number of VMs needed in Azure might not equate to what you currently have deployed in an on-premises datacenter. Make sure you choose the right size for the workloads that you plan to run.

### Use purchase discounts

Azure has many discounts that your organization should take advantage of to save money.

### Azure Reservations

Azure Reservations allow you to prepay for one-year or three-years of virtual machine or SQL Database compute capacity. Pre-paying will allow you to get a discount on the resources you use. Azure reservations can significantly reduce your virtual machine or SQL database compute costs — up to 72 percent on pay-as-you-go prices with one-year or three-year upfront commitment. Reservations provide a billing discount and don't affect the runtime state of your virtual machines or SQL databases.

## Use Azure Hybrid Benefit

If you already have Windows Server or SQL Server licenses in your on-premises deployments, you can use the Azure Hybrid Benefit program to save in Azure. With the Windows Server benefit, each license covers the cost of the OS (up to two virtual machines), and you only pay for base compute costs. You can use existing SQL Server licenses to save up to 55 percent on vCore-based SQL Database options. Options include SQL Server in Azure Virtual Machines and SQL Server Integration Services.

## Other resources

Azure also has a service that allows you to build services that take advantage of surplus capacity in Azure for reduced rates.

# Azure Managed Applications

Azure managed applications enable you to offer cloud solutions that are easy for consumers to deploy and operate. You implement the infrastructure and provide ongoing support. To make a managed application available to all customers, publish it in the Azure marketplace. To make it available to only users in your organization, publish it to an internal catalog.

A managed application is similar to a solution template in the Marketplace, with one key difference. In a managed application, the resources are deployed to a resource group that's managed by the publisher of the app. The resource group is present in the consumer's subscription, but an identity in the publisher's tenant has access to the resource group. As the publisher, you specify the cost for ongoing support of the solution.

## Advantages of managed applications

Managed applications reduce barriers to consumers using your solutions. They don't need expertise in cloud infrastructure to use your solution. Consumers have limited access to the critical resources, and don't need to worry about making a mistake when managing it.

Managed applications enable you to establish an ongoing relationship with your consumers. You define terms for managing the application, and all charges are handled through Azure billing.

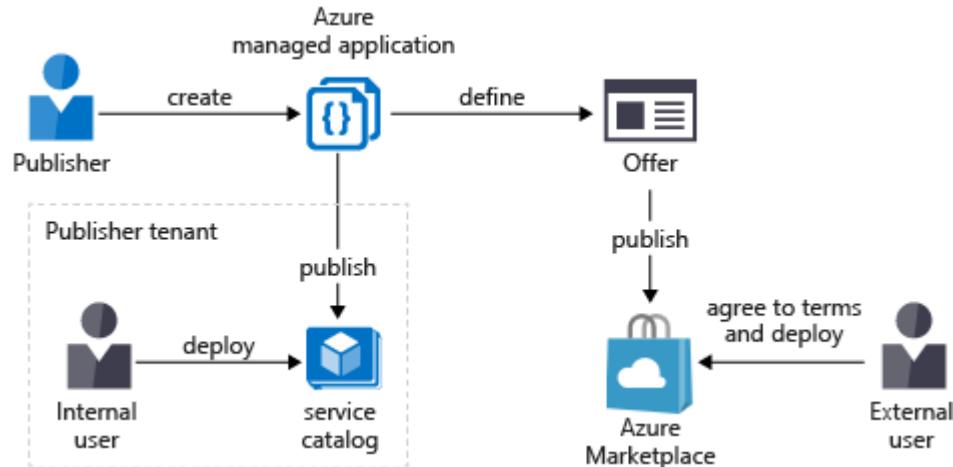
Although customers deploy these managed applications in their subscriptions, they don't have to maintain, update, or service them. You can make sure that all customers are using approved versions. Customers don't have to develop application-specific domain knowledge to manage these applications. Customers automatically acquire application updates without the need to worry about troubleshooting and diagnosing issues with the applications.

For IT teams, managed applications enable you to offer pre-approved solutions to users in the organization. You know these solutions are compliant with organizational standards.

Managed Applications support managed identities for Azure resources.

## Types of managed applications

You can publish your managed application either externally or internally.



## Service catalog

The service catalog is an internal catalog of approved solutions for users in an organization. You use the catalog to meet organizational standards while offering solutions for the organizations. Employees use the catalog to easily find applications that are recommended and approved by their IT departments. They see the managed applications that other people in their organization share with them.

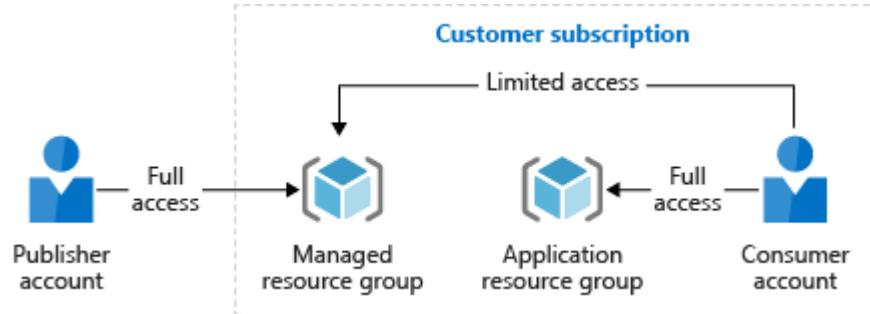
## Marketplace

Vendors wishing to bill for their services can make a managed application available through the Azure marketplace. After the vendor publishes an application, it's available to users outside the organization. With this approach, managed service providers (MSPs), independent software vendors (ISVs), and system integrators (SIs) can offer their solutions to all Azure customers.

## Resource groups for managed applications

Typically, the resources for a managed application are in two resource groups. The consumer manages one resource group, and the publisher manages the other resource group. When defining the managed application, the publisher specifies the levels of access. The publisher can request either a permanent role assignment, or just-in-time access for an assignment that is constrained to a time period.

Restricting access for data operations is currently not supported for all data providers in Azure. The following image shows a scenario where the publisher requests the owner role for the managed resource group. The publisher placed a read-only lock on this resource group for the consumer. The publisher's identities that are granted access to the managed resource group are exempt from the lock.



## Application resource group

This resource group holds the managed application instance. This resource group may only contain one resource. The resource type of the managed application is Microsoft.Solutions/applications.

The consumer has full access to the resource group and uses it to manage the lifecycle of the managed application.

## Managed resource group

This resource group holds all the resources that are required by the managed application. For example, this resource group contains the virtual machines, storage accounts, and virtual networks for the solution. The consumer has limited access to this resource group because the consumer doesn't manage the individual resources for the managed application. The publisher's access to this resource group corresponds to the role specified in the managed application definition. For example, the publisher might request the Owner or Contributor role for this resource group. The access is either permanent or limited to a specific time.

When publishing the managed application to the marketplace, the publisher can grant consumers the ability to perform specific actions on resources in the managed resource group.

For example, the publisher can specify that consumers can restart virtual machines. All other actions beyond read actions are still denied. Changes to resources in a managed resource group by a consumer with granted actions are subject to the Azure Policy assignments within the consumers tenant scoped to include the managed resource group.

When the consumer deletes the managed application, the managed resource group is also deleted.

## Azure Policy

You can apply an Azure Policy to audit your managed application. You apply policy definitions to make sure deployed instances of your managed application fulfill data and

security requirements. If your application interacts with sensitive data, make sure you've evaluated how that should be protected. For example, if your application interacts with data from Microsoft 365, apply a policy definition to make sure data encryption is enabled.

## View definition artifact in Azure Managed Applications

View definition is an optional artifact in Azure Managed Applications. It allows to customize overview page and add more views such as Metrics and Custom resources.

### View definition artifact

The view definition artifact must be named `viewDefinition.json` and placed at the same level as `createUiDefinition.json` and `mainTemplate.json` in the .zip package that creates a managed application definition.

### View definition schema

The `viewDefinition.json` file has only one top level `views` property, which is an array of views. Each view is shown in the managed application user interface as a separate menu item in the table of contents. Each view has a `kind` property that sets the type of the view. It must be set to one of the following values: `Overview`, `Metrics`, `CustomResources`, `Associations`.

Sample JSON for view definition:

```
{  
    "$schema": "https://schema.management.azure.com/schemas/viewdefinition/0.0.1-preview/ViewDefinition.json#",  
    "contentVersion": "0.0.0.1",  
    "views": [  
        {  
            "kind": "Overview",  
            "properties": {  
                "header": "Welcome to your Azure Managed Application",  
                "description": "This managed application is for demo purposes only.",  
                "commands": [  
                    {  
                        "displayName": "Test Action",  
                        "path": "testAction"  
                    }  
                ]  
            },  
            {  
                "kind": "Metrics",  
                "properties": {  
                    "title": "Metrics Overview",  
                    "description": "A summary of key metrics for this managed application."  
                }  
            }  
        }  
    ]  
}
```

```

"properties": {
    "displayName": "This is my metrics view",
    "version": "1.0.0",
    "charts": [
        {
            "displayName": "Sample chart",
            "chartType": "Bar",
            "metrics": [
                {
                    "name": "Availability",
                    "aggregationType": "avg",
                    "resourceTagFilter": [ "tag1" ],
                    "resourceType": "Microsoft.Storage/storageAccounts",
                    "namespace": "Microsoft.Storage/storageAccounts"
                }
            ]
        }
    ]
},
{
    "kind": "CustomResources",
    "properties": {
        "displayName": "Test custom resource type",
        "version": "1.0.0",
        "resourceType": "testCustomResource",
        "createUIDefinition": { },
        "commands": [
            {
                "displayName": "Custom Context Action",
                "path": "testCustomResource/testContextAction",
                "icon": "Stop",
                "createUIDefinition": { }
            }
        ],
        "columns": [
            {"key": "name", "displayName": "Name"},
            {"key": "properties.myProperty1", "displayName": "Property 1"},
            {"key": "properties.myProperty2", "displayName": "Property 2", "optional": true}
        ]
    }
},

```

```
{
  "kind": "Associations",
  "properties": {
    "displayName": "Test association resource type",
    "version": "1.0.0",
    "targetResourceType": "Microsoft.Compute/virtualMachines",
    "createUIDefinition": { }
  }
}
]
```

## Overview

"kind": "Overview"

When you provide this view in viewDefinition.json, it overrides the default Overview page in your managed application.

```
{
  "kind": "Overview",
  "properties": {
    "header": "Welcome to your Azure Managed Application",
    "description": "This managed application is for demo purposes only.",
    "commands": [
      {
        "displayName": "Test Action",
        "path": "testAction"
      }
    ]
  }
}
```

Property	Required	Description
----------	----------	-------------

---

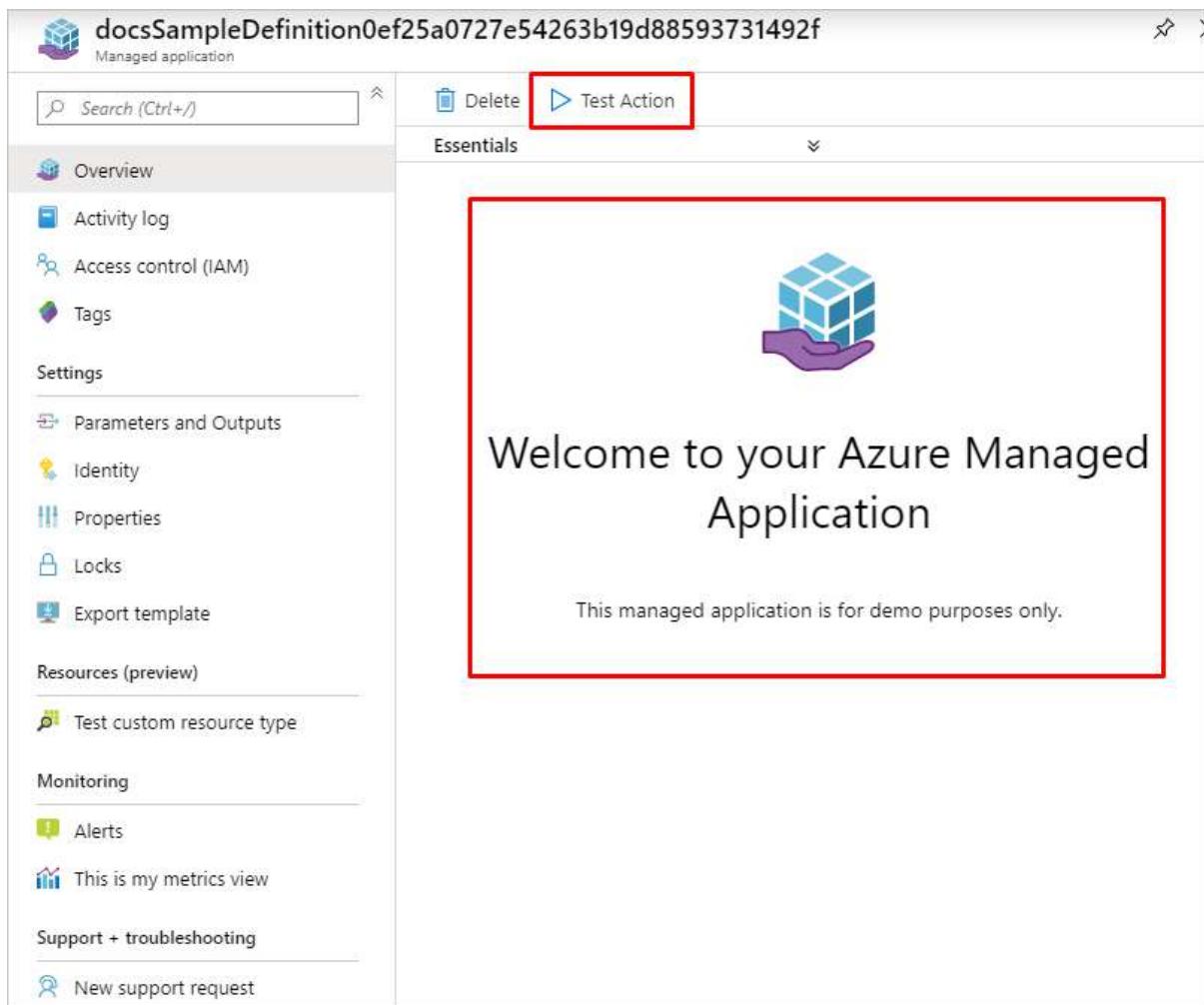
header	No	The header of the overview page.
--------	----	----------------------------------

---

description	No	The description of your managed application.
-------------	----	--

---

command	No	The array of additional toolbar buttons of the overview page.
---------	----	---



## Metrics

"kind": "Metrics"

The metrics view enables you to collect and aggregate data from your managed application resources in Azure Monitor Metrics.

```
{
  "kind": "Metrics",
  "properties": {
    "displayName": "This is my metrics view",
    "version": "1.0.0",
    "charts": [
      {
        "displayName": "Sample chart",
        "chartType": "Bar",
        "metrics": [
          ...
        ]
      }
    ]
  }
}
```

```

    {
      "name": "Availability",
      "aggregationType": "avg",
      "resourceTagFilter": [ "tag1" ],
      "resourceType": "Microsoft.Storage/storageAccounts",
      "namespace": "Microsoft.Storage/storageAccounts"
    }
  ]
}
]
}

```

Property	Required	Description
----------	----------	-------------

---

displayName	No	The displayed title of the view.
-------------	----	----------------------------------

---

version	No	The version of the platform used to render the view.
---------	----	--

---

charts	Yes	The array of charts of the metrics page.
--------	-----	--

## Chart

Property	Required	Description
----------	----------	-------------

---

displayName	Yes	The displayed title of the chart.
-------------	-----	-----------------------------------

---

chartType	No	The visualization to use for this chart. By default, it uses a line chart. Supported chart types: Bar, Line, Area, Scatter.
-----------	----	---

---

metrics	Yes	The array of metrics to plot on this chart.
---------	-----	---

## Metric

---

Property	Required	Description
----------	----------	-------------

---

name	Yes	The name of the metric.
------	-----	-------------------------

---

aggregationType	Yes	The aggregation type to use for this metric. Supported aggregation types: none, sum, min, max, avg, unique, percentile, count
-----------------	-----	---

---

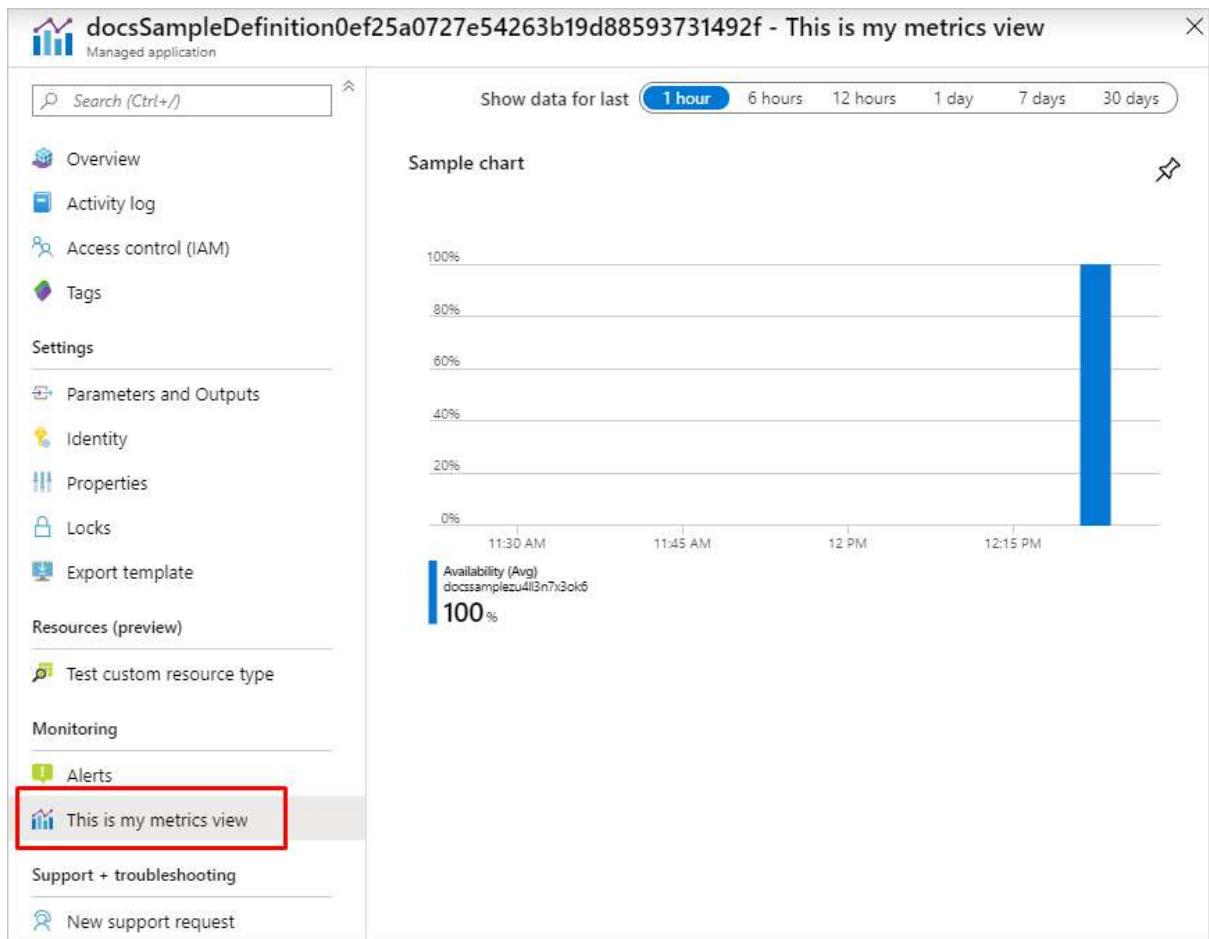
namespace	No	Additional information to use when determining the correct metrics provider.
-----------	----	--

---

resourceTagFilter	No	The resource tags array (will be separated with or word) for which metrics would be displayed. Applies on top of resource type filter.
-------------------	----	--

---

resourceType	Yes	The resource type for which metrics would be displayed.
--------------	-----	---



## Custom resources

"kind": "CustomResources"

You can define multiple views of this type. Each view represents a unique custom resource type from the custom provider you defined in mainTemplate.json.

In this view you can perform GET, PUT, DELETE and POST operations for your custom resource type. POST operations could be global custom actions or custom actions in a context of your custom resource type.

JSON

Copy

{

```

"kind": "CustomResources",
"properties": {
    "displayName": "Test custom resource type",
    "version": "1.0.0",
    "resourceType": "testCustomResource",
    "icon": "Polychromatic.ResourceList",
    "createUIDefinition": { },
    "commands": [
        {

```

```

    "displayName": "Custom Context Action",
    "path": "testCustomResource/testContextAction",
    "icon": "Stop",
    "createUIDefinition": { },
}
],
"columns": [
    {"key": "name", "displayName": "Name"},
    {"key": "properties.myProperty1", "displayName": "Property 1"},
    {"key": "properties.myProperty2", "displayName": "Property 2", "optional": true}
]
}
}

```

Property	Required	Description
----------	----------	-------------

---

displayName	Yes	The displayed title of the view. The title should be unique for each CustomResources view in your viewDefinition.json.
-------------	-----	--

---

version	No	The version of the platform used to render the view.
---------	----	--

---

resourceType	Yes	The custom resource type. Must be a unique custom resource type of your custom provider.
--------------	-----	--

---

icon	No	The icon of the view.
------	----	-----------------------

---

createUIDefinition	No	Create UI Definition schema for create custom resource command.
--------------------	----	---

---

commands	No	The array of additional toolbar buttons of the CustomResources view.
columns	No	The array of columns of the custom resource. If not defined the name column will be shown by default. The column must have "key" and "displayName". For key, provide the key of the property to display in a view. If nested, use dot as delimiter, for example, "key": "name" or "key": "properties.property1". For display name, provide the display name of the property to display in a view. You can also provide an "optional" property. When set to true, the column is hidden in a view by default.

The screenshot shows the Azure portal interface for a custom resource type named 'Test custom resource type'. The left sidebar includes links for Overview, Activity log, Access control (IAM), Tags, Parameters and Outputs, Identity, Properties, Locks, Export template, and Monitoring. The main content area displays the resource details, with a red box highlighting the 'Test custom resource type' card under 'Resources (preview)'. The toolbar at the top right contains buttons for Add, Edit columns, Refresh, Remove, and a 'Custom Context Action' button, which is also highlighted with a red box.

## Commands

Commands is an array of additional toolbar buttons that are displayed on page. Each command represents a POST action from your Azure Custom Provider defined in mainTemplate.json.

```
{
  "commands": [
    {
      "label": "Label for the first command"
    }
  ]
}
```

```

    "displayName": "Start Test Action",
    "path": "testAction",
    "icon": "Start",
    "createUIDefinition": {}
},
]
}

```

Property	Required	Description
----------	----------	-------------

---

displayName	Yes	The displayed name of the command button.
-------------	-----	---

---

path	Yes	The custom provider action name. The action must be defined in mainTemplate.json.
------	-----	---

---

icon	No	The icon of the command button.
------	----	---------------------------------

---

createUIDefinition	No	Create UI Definition schema for command.
--------------------	----	--

## Associations

"kind": "Associations"

You can define multiple views of this type. This view allows you to link existing resources to the managed application through the custom provider you defined in mainTemplate.json.

In this view you can extend existing Azure resources based on the targetResourceType. When a resource is selected, it will create an onboarding request to the public custom provider, which can apply a side effect to the resource.

```
{
  "kind": "Associations",
  "properties": {
    "displayName": "Test association resource type",
    "path": "testAssociation"
  }
}
```

```

    "version": "1.0.0",
    "targetResourceType": "Microsoft.Compute/virtualMachines",
    "createUIDefinition": { }
}
}

```

Property	Required	Description
displayName	Yes	The displayed title of the view. The title should be unique for each Associations view in your viewDefinition.json.
version	No	The version of the platform used to render the view.
targetResource Type	Yes	The target resource type. This is the resource type that will be displayed for resource onboarding.
createUIDefinition	No	Create UI Definition schema for create association resource command.

## Azure security baseline for Azure Managed Applications

This security baseline applies guidance from the Azure Security Benchmark version 2.0 to Azure Managed Applications. The Azure Security Benchmark provides recommendations on how you can secure your cloud solutions on Azure. The content is grouped by the security controls defined by the Azure Security Benchmark and the related guidance applicable to Azure Managed Applications.

**Note:** Controls not applicable to Azure Managed Applications, and those for which the global guidance is recommended verbatim, have been excluded.

## **Network Security**

NS-1: Implement security for internal traffic

Guidance: Not applicable; Azure Managed Applications defines the deployment of Azure services that could have a networking component, but does not have one itself. Please refer to each resource's individual documentation for its security information.

Responsibility: Microsoft

Microsoft Defender for Cloud monitoring: None

NS-6: Simplify network security rules

Guidance: Use Azure Virtual Network Service Tags to define network access controls on network security groups or Azure Firewall configured for your Azure Managed Application resources. You can use service tags in place of specific IP addresses when creating security rules. By specifying the service tag name (For example: AzureResourceManager) in the appropriate source or destination field of a rule, you can allow or deny the traffic for the corresponding service. Microsoft manages the address prefixes encompassed by the service tag and automatically updates the service tag as addresses change.

Responsibility: Customer

Microsoft Defender for Cloud monitoring: None

NS-7: Secure Domain Name Service (DNS)

Guidance: Configurations related to DNS for the Managed Applications service are maintained by Microsoft. Azure Managed Applications defines the deployment of Azure services that could expose its underlying DNS configurations, but does not itself. Please refer to each resource's individual documentation for its security information.

Responsibility: Microsoft

Microsoft Defender for Cloud monitoring: None

## **Identity Management**

IM-1: Standardize Azure Active Directory as the central identity and authentication system

Guidance: Azure Managed Applications uses Azure Active Directory (Azure AD) as the default identity and access management service. Standardize on Azure AD to govern your organization's identity and access management.

Azure provides the following Azure built-in roles for authorizing access to Managed Applications using Azure AD and OAuth:

- Managed Application Contributor Role: Allows for creating managed application resources.

- Managed Application Operator Role: Lets you read and perform actions on Managed Application resources
- Managed Applications Reader: Lets you read resources in a managed app and request JIT access.

Responsibility: Customer

Microsoft Defender for Cloud monitoring: None

#### IM-2: Manage application identities securely and automatically

Guidance: Use Azure-managed identities for granting permissions to your Managed Applications. It is recommended to use Azure managed identity feature instead of creating a more powerful human account to access or execute your resources to limit the need to manage additional credentials. The Azure Managed Applications service can also be assigned a managed identity itself to natively authenticate to other Azure services/resources that supports Azure Active Directory (Azure AD) authentication. This can be useful to enable easy access from your Azure Managed Applications to Azure Key Vault when retrieving secrets. When using managed identities, the identity is managed by the Azure platform and does not require you to provision or rotate any secrets.

Azure Managed Applications supports your application being granted two types of identities:

- A system-assigned identity is tied to your configuration resource. It's deleted if your configuration resource is deleted. A configuration resource can only have one system-assigned identity.
- A user-assigned identity is a standalone Azure resource that can be assigned to your configuration resource. A configuration resource can have multiple user-assigned identities.

When managed identities cannot be leveraged, create a service principal with restricted permissions at the Azure Managed Application resource level. Configure these service principals with certificate credentials and only fall back to client secrets. In both cases, Azure Key Vault can be used to in conjunction with Azure managed identities, so that the runtime environment (e.g., an Azure function) can retrieve the credential from the key vault.

Responsibility: Customer

Microsoft Defender for Cloud monitoring: None

#### IM-3: Use Azure AD single sign-on (SSO) for application access

Guidance: Azure Managed Applications use Azure Active Directory (Azure AD) to provide identity and access management to Azure resources, cloud applications, and on-premises applications. This includes enterprise identities such as employees, as well as external identities such as partners, vendors, and suppliers. This enables single sign-on (SSO) to manage and secure access to your organization's data and resources on-premises and in the cloud. Connect all your users, applications, and devices to the Azure AD for seamless, secure access and greater visibility and control.

Responsibility: Customer

Microsoft Defender for Cloud monitoring: None

## Privileged Access

PA-1: Protect and limit highly privileged users

Guidance: Azure Managed Applications uses Azure Active Directory (Azure AD) for identity and access. The most critical built-in roles are Azure AD are Global Administrator and the Privileged Role Administrator as users assigned to these two roles can delegate administrator roles:

- Global Administrator: Users with this role have access to all administrative features in Azure AD, as well as services that use Azure AD identities.
- Privileged Role Administrator: Users with this role can manage role assignments in Azure AD, as well as within Azure AD Privileged Identity Management (PIM). In addition, this role allows management of all aspects of PIM and administrative units.

Note: You may have other critical roles that need to be governed if you use custom roles with certain privileged permissions assigned. And you may also want to apply similar controls to the administrator account of critical business assets.

You can enable just-in-time (JIT) privileged access to Azure resources and Azure AD using Azure AD Privileged Identity Management (PIM). JIT grants temporary permissions to perform privileged tasks only when users need it. PIM can also generate security alerts when there is suspicious or unsafe activity in your Azure AD organization.

Responsibility: Customer

Microsoft Defender for Cloud monitoring: None

PA-3: Review and reconcile user access regularly

Guidance: Review user accounts and access assignment regularly to ensure the accounts and their level of access are valid.

Azure Managed Applications uses Azure Active Directory (Azure AD) accounts to manage its resources, review user accounts and access assignment regularly to ensure the accounts and their access are valid. You can use Azure AD access reviews to review group memberships, access to enterprise applications, and role assignments. Azure AD reporting can provide logs to help discover stale accounts. You can also use Azure AD Privileged Identity Management to create access review report workflow to facilitate the review process. In addition, Azure Privileged Identity Management can also be configured to alert when an excessive number of administrator accounts are created, and to identify administrator accounts that are stale or improperly configured.

Note: Some Azure services support local users and roles which not managed through Azure AD. You will need to manage these users separately.

Responsibility: Customer

Microsoft Defender for Cloud monitoring: None

## PA-6: Use privileged access workstations

Guidance: Secured, isolated workstations are critically important for the security of sensitive roles like administrators, developers, and critical service operators. Use highly secured user workstations and/or Azure Bastion for administrative tasks related to your Managed Applications. Use Azure Active Directory (Azure AD), Microsoft Defender Advanced Threat Protection (ATP), and/or Microsoft Intune to deploy a secure and managed user workstation for administrative tasks. The secured workstations can be centrally managed to enforce secured configuration including strong authentication, software and hardware baselines, restricted logical and network access.

Responsibility: Customer

Microsoft Defender for Cloud monitoring: None

## PA-7: Follow just enough administration (least privilege principle)

Guidance: Azure Managed Applications is integrated with Azure role-based access control (RBAC) to manage its resources. Azure RBAC allows you to manage Azure resource access through role assignments. You can assign these roles to users, groups service principals and managed identities. There are pre-defined built-in roles for certain resources, and these roles can be inventoried or queried through tools such as Azure CLI, Azure PowerShell or the Azure portal. The privileges you assign to resources through the Azure RBAC should be always limited to what is required by the roles. This complements the just in time (JIT) approach of Azure Active Directory (Azure AD) Privileged Identity Management (PIM) and should be reviewed periodically.

Where possible, use built-in roles to allocate permission and only create custom role when required.

Azure provides the following Azure built-in roles for authorizing access to Managed Applications using Azure AD and OAuth:

- Managed Application Contributor Role: Allows for creating managed application resources.
- Managed Application Operator Role: Lets you read and perform actions on Managed Application resources
- Managed Applications Reader: Lets you read resources in a managed app and request JIT access.

Responsibility: Customer

Microsoft Defender for Cloud monitoring: None

## PA-8: Choose approval process for Microsoft support

Guidance: Implement an organizational approval process for support scenarios where Microsoft may need access to your Azure Managed Application data. Customer Lockbox is not currently available for Managed Application scenarios.

Responsibility: Customer

Microsoft Defender for Cloud monitoring: None

## Data Protection

DP-2: Protect sensitive data

Guidance: For bringing encryption with your own keys, you can utilize your own storage account for the storage of the Managed Application configuration files.

Responsibility: Customer

Microsoft Defender for Cloud monitoring: None

DP-4: Encrypt sensitive information in transit

Guidance: Azure Managed Applications is an Azure service that utilizes Azure Resource Manager for all service actions. HTTPS/TLS is used to protect data in transit for Azure Resource Manager.

Responsibility: Microsoft

Microsoft Defender for Cloud monitoring: None

DP-5: Encrypt sensitive data at rest

Guidance: Managed application definitions which define your application and its resources can be stored in your own storage accounts which can be configured with customer-managed encryption keys.

For scenarios where you do not want to bring your own storage for Managed Application definitions, Azure provides data at rest encryption by default.

Responsibility: Shared

Microsoft Defender for Cloud monitoring: None

## Asset Management

AM-1: Ensure security team has visibility into risks for assets

Guidance: Ensure security teams are granted Security Reader permissions in your Azure tenant and subscriptions so they can monitor for security risks using Microsoft Defender for Cloud.

Depending on how security team responsibilities are structured, monitoring for security risks could be the responsibility of a central security team or a local team. That said, security insights and risks must always be aggregated centrally within an organization.

Security Reader permissions can be applied broadly to an entire tenant (Root Management Group) or scoped to management groups or specific subscriptions.

Note: Additional permissions might be required to get visibility into workloads and services.

**Responsibility:** Customer

**Microsoft Defender for Cloud monitoring:** None

**AM-2:** Ensure security team has access to asset inventory and metadata

**Guidance:** Apply tags to your Azure resources, resource groups, and subscriptions to logically organize them into a taxonomy. Each tag consists of a name and a value pair. For example, you can apply the name "Environment" and the value "Production" to all the resources in production.

Tags that are provided at creation time of the managed app are applied to the Managed Resource Group as well. The publisher of your application can provide their own additional tagging of the managed resources after deployment.

**Responsibility:** Customer

**Microsoft Defender for Cloud monitoring:** None

**AM-3:** Use only approved Azure services

**Guidance:** Azure Managed Applications supports Azure Resource Manager based deployments and configuration enforcement using Azure Policy. Use Azure Policy to audit and restrict which services users can provision in your environment. Use Azure Resource Graph to query for and discover resources within their subscriptions. You can also use Azure Monitor to create rules to trigger alerts when a non-approved service is detected.

**Responsibility:** Customer

**Microsoft Defender for Cloud monitoring:** None

## **Logging and Threat Detection**

**LT-2:** Enable threat detection for Azure identity and access management

**Guidance:** Azure Active Directory (Azure AD) provides the following user logs that can be viewed in Azure AD reporting or integrated with Azure Monitor, Microsoft Sentinel or other SIEM/monitoring tools for more sophisticated monitoring and analytics use cases:

**Sign-ins** – The sign-ins report provides information about the usage of managed applications and user sign-in activities.

**Audit logs** - Provides traceability through logs for all changes done by various features within Azure AD. Examples of audit logs include changes made to any resources within Azure AD like adding or removing users, apps, groups, roles and policies.

**Risky sign-ins** - A risky sign-in is an indicator for a sign-in attempt that might have been performed by someone who is not the legitimate owner of a user account.

**Users flagged for risk** - A risky user is an indicator for a user account that might have been compromised.

Microsoft Defender for Cloud can also alert on certain suspicious activities such as excessive number of failed authentication attempts, deprecated accounts in the subscription. In addition

to the basic security hygiene monitoring, Microsoft Defender for Cloud's Threat Protection module can also collect more in-depth security alerts from individual Azure compute resources (virtual machines, containers, app service), data resources (SQL DB and storage), and Azure service layers. This capability gives you visibility on account anomalies inside the individual resources.

**Responsibility:** Customer

**Microsoft Defender for Cloud monitoring:** None

#### LT-4: Enable logging for Azure resources

**Guidance:** Activity logs, which are automatically available, contain all write operations (PUT, POST, DELETE) for your managed application resources except read operations (GET). Activity logs can be used to find an error when troubleshooting or to monitor how a user in your organization modified a resource.

**Responsibility:** Customer

**Microsoft Defender for Cloud monitoring:** None

#### LT-5: Centralize security log management and analysis

**Guidance:** Centralize logging, storage, and analysis to enable correlation. For each log source, ensure you have assigned a data owner, access guidance, storage location, what tools are used to process and access the data, and data retention requirements. Ensure you are integrating Azure activity logs into your central logging. Ingest logs via Azure Monitor to aggregate security data generated by endpoint devices, network resources, and other security systems. In Azure Monitor, use Log Analytics workspaces to query and perform analytics, and use Azure Storage accounts for long term and archival storage.

In addition, enable and onboard data to Microsoft Sentinel or a third-party SIEM.

Many organizations choose to use Microsoft Sentinel for “hot” data that is used frequently and Azure Storage for “cold” data that is used less frequently.

**Responsibility:** Customer

**Microsoft Defender for Cloud monitoring:** None

#### LT-6: Configure log storage retention

**Guidance:** Ensure that any storage accounts or Log Analytics workspaces used for storing logs created by your managed application resources has the log retention period set according to your organization's compliance regulations. In Azure Monitor, you can set your Log Analytics workspace retention period according to your organization's compliance regulations. Use Azure Storage, Data Lake or Log Analytics workspace accounts for long-term and archival storage.

**Responsibility:** Customer

**Microsoft Defender for Cloud monitoring:** None

## LT-7: Use approved time synchronization sources

Guidance: Azure Managed Applications defines the deployment of Azure services that could support configuring your own time synchronization sources, but does not itself. The Managed Applications service relies on Microsoft time synchronization sources, and is not exposed to customers for configuration. Please refer to each resource's individual documentation for its security information.

Responsibility: Microsoft

Microsoft Defender for Cloud monitoring: None

## Posture and Vulnerability Management

### PV-1: Establish secure configurations for Azure services

Guidance: Define security guardrails for infrastructure and DevOps teams by making it easy to securely configure the Azure services they use.

Configure Azure Policy to audit and enforce configurations of your resources related to your Managed Applications deployments.

You can use Azure Blueprints to automate deployment and configuration of services and application environments, including Azure Resource Manager templates, Azure RBAC assignments, and Azure Policy assignments, in a single blueprint definition.

Responsibility: Customer

Microsoft Defender for Cloud monitoring: None

### PV-2: Sustain secure configurations for Azure services

Guidance: Use Microsoft Defender for Cloud to monitor your resources related to Azure Managed Applications and use Azure Policy [deny] and [deploy if not exist] effects to sustain secure configurations.

Responsibility: Customer

Microsoft Defender for Cloud monitoring: None

### PV-6: Perform software vulnerability assessments

Guidance: Azure Managed Applications defines the deployment of Azure services, please refer to each resource's individual documentation for its security information. The Managed Applications service does not expose customer-facing compute resources which would support vulnerability assessment tools.

Microsoft handles vulnerabilities and assessments for the underlying platform that supports the Managed Applications service.

Responsibility: Microsoft

Microsoft Defender for Cloud monitoring: None

## PV-7: Rapidly and automatically remediate software vulnerabilities

Guidance: Azure Managed Applications defines the deployment of Azure services, please refer to each resource's individual documentation for its security information. The Managed Applications service does not expose customer-facing compute resources which would support vulnerability assessment tools.

Microsoft handles vulnerabilities and assessments for the underlying platform that supports the Managed Applications service.

Responsibility: Microsoft

Microsoft Defender for Cloud monitoring: None

## PV-8: Conduct regular attack simulation

Guidance: As required, conduct penetration testing or red team activities on your Azure resources and ensure remediation of all critical security findings. Follow the Microsoft Cloud Penetration Testing Rules of Engagement to ensure your penetration tests are not in violation of Microsoft policies. Use Microsoft's strategy and execution of Red Teaming and live site penetration testing against Microsoft-managed cloud infrastructure, services, and applications.

Responsibility: Shared

Microsoft Defender for Cloud monitoring: None

# Endpoint Security

## ES-1: Use Endpoint Detection and Response (EDR)

Guidance: Azure Managed Applications defines the deployment of Azure services that could interact with virtual machines, containers, or storage which requires Endpoint Detection and Response (EDR) protection, but does not itself. Please refer to each resource's individual documentation for its security information.

The underlying infrastructure for Azure Managed Applications is handled by Microsoft, which includes anti-malware and EDR handling.

Responsibility: Microsoft

Microsoft Defender for Cloud monitoring: None

## ES-2: Use centrally managed modern anti-malware software

Guidance: Azure Managed Applications defines the deployment of Azure services that could interact with virtual machines, containers, or storage which require anti-malware protection, but does not itself. Please refer to each resource's individual documentation for its security information.

The underlying infrastructure for Azure Managed Applications is handled by Microsoft, which includes anti-malware and EDR handling.

Responsibility: Microsoft

Microsoft Defender for Cloud monitoring: None

ES-3: Ensure anti-malware software and signatures are updated

Guidance: Azure Managed Applications defines the deployment of Azure services that could interact with virtual machines, containers, or storage which require anti-malware protection, but does not itself. Please refer to each resource's individual documentation for its security information.

The underlying infrastructure for Azure Managed Applications is handled by Microsoft, which includes anti-malware and EDR handling.

Responsibility: Microsoft

Microsoft Defender for Cloud monitoring: None

## **Backup and Recovery**

BR-3: Validate all backups including customer-managed keys

Guidance: When storing your Managed Application definitions in your own storage account, ensure that you can restore any associated customer-managed keys that are used for that account's encryption which are stored in Azure Key Vault.

Responsibility: Customer

Microsoft Defender for Cloud monitoring: None

BR-4: Mitigate risk of lost keys

Guidance: If you are bringing your own storage for your Managed Application definitions, ensure you have measures in place to prevent and recover from loss of keys used to encrypt your definitions. Enable soft delete and purge protection on the Azure Key Vault which stores your customer-managed keys to protect keys against accidental or malicious deletion.

Responsibility: Customer

Microsoft Defender for Cloud monitoring: None

## **Azure Scheduler**

### **What is Azure Scheduler?**

Azure Scheduler helps you create jobs that run in the cloud by declaratively describing actions. The service then automatically schedules and runs those actions. For example, you can call services inside and outside Azure, such as calling HTTP or HTTPS endpoints, and also post messages to Azure Storage queues and Azure Service Bus queues or topics. You can

run jobs immediately or at a later time. Scheduler easily supports complex schedules and advanced recurrence. Scheduler specifies when to run jobs, keeps a history of job results that you can review, and then predictably and reliably schedules workloads to run.

Other Azure scheduling capabilities also use Scheduler in the background, for example, Azure WebJobs, which is a Web Apps feature in Azure App Service. You can manage communication for these actions by using the Scheduler REST API, which helps you manage the communication for these actions.

Here are some scenarios where Scheduler can help you:

- Run recurring app actions: For example, periodically collect data from Twitter into a feed.
- Perform daily maintenance: Such as pruning logs daily, performing backups, and other maintenance tasks.

For example, as an administrator, you might want to back up your database at 1:00 AM every day for the next nine months.

Although you can use Scheduler to create, maintain, and run scheduled workloads, Scheduler doesn't host the workloads or run code. The service only invokes the services or code hosted elsewhere, for example, in Azure, on-premises, or with another provider. Scheduler can invoke through HTTP, HTTPS, a Storage queue, a Service Bus queue, or a Service Bus topic. To create, schedule, manage, update, or delete jobs and job collections, you can use code, the Scheduler REST API, or the Azure Scheduler PowerShell cmdlets.

## Concepts, terminology, and entities in Azure Scheduler

### Entity hierarchy

The Azure Scheduler REST API exposes and uses these main entities, or resources:

Entity	Description
Job	Defines a single recurring action with simple or complex strategies for execution. Actions might include HTTP, Storage queue, Service Bus queue, or Service Bus topic requests.
Job collection	Contains a group of jobs and maintains settings, quotas, and throttles that are shared by jobs in the collection. As an Azure subscription owner, you can create job collections and group jobs together based on their usage or application boundaries. A job collection has these attributes: <ul style="list-style-type: none"><li>- Constrained to one region.</li><li>- Lets you enforce quotas so you can constrain usage for all jobs in a collection.</li><li>- Quotas include MaxJobs and MaxRecurrence.</li></ul>

---

Job history	Describes details for a job execution, for example, status and any response details.
-------------	--

---

## Entity management

At a high-level, the Scheduler REST API exposes these operations for managing entities.

### Job management

Supports operations for creating and editing jobs. All jobs must belong to an existing job collection, so there's no implicit creation. Here's the URI address for these operations:

`https://management.azure.com/subscriptions/{subscriptionID}/resourceGroups/{resourceGroupName}/providers/Microsoft.Scheduler/jobCollections/{jobCollectionName}/jobs/{jobName}`

### Job collection management

Supports operations for creating and editing jobs and job collections, which map to quotas and shared settings. For example, quotas specify the maximum number of jobs and smallest recurrence interval. Here's the URI address for these operations:

`https://management.azure.com/subscriptions/{subscriptionID}/resourceGroups/{resourceGroupName}/providers/Microsoft.Scheduler/jobCollections/{jobCollectionName}`

### Job history management

Supports the GET operation for fetching 60 days of job execution history, for example, job elapsed time and job execution results. Includes query string parameter support for filtering based on state and status. Here's the URI address for this operation:

`https://management.azure.com/subscriptions/{subscriptionID}/resourceGroups/{resourceGroupName}/providers/Microsoft.Scheduler/jobCollections/{jobCollectionName}/jobs/{jobName}/history`

## Job types

Azure Scheduler supports multiple job types:

- HTTP jobs, including HTTPS jobs that support TLS, for when you have the endpoint for an existing service or workload
- Storage queue jobs for workloads that use Storage queues, such as posting messages to Storage queues
- Service Bus queue jobs for workloads that use Service Bus queues
- Service Bus topic jobs for workloads that use Service Bus topics

## Job definition

At the high level, a Scheduler job has these basic parts:

- The action that runs when the job timer fires
- Optional: The time to run the job
- Optional: When and how often to repeat the job
- Optional: An error action that runs if the primary action fails

The job also includes system-provided data such as the job's next scheduled run time. The job's code definition is an object in JavaScript Object Notation (JSON) format, which has these elements:

Element	Required	Description
startTime	No	The start time for the job with a time zone offset in ISO 8601 format
action	Yes	The details for the primary action, which can include an errorAction object
errorAction	No	The details for the secondary action that runs if the primary action fails
recurrence	No	The details such as frequency and interval for a recurring job
retryPolicy	No	The details for how often to retry an action
state	Yes	The details for the job's current state
status	Yes	The details for the job's current status, which is controlled by the service

Here's an example that shows a comprehensive job definition for an HTTP action with fuller element details described in later sections:

```
"properties": {
  "startTime": "2012-08-04T00:00Z",
  "action": {
    "type": "Http",
    "request": {
      "uri": "http://contoso.com/some-method",
      "method": "PUT",
      "body": "Posting from a timer",
      "headers": {
        "Content-Type": "application/json"
      }
    }
  }
}
```

```

},
"retryPolicy": {
    "retryType": "None"
},
},
"errorAction": {
    "type": "Http",
    "request": {
        "uri": "http://contoso.com/notifyError",
        "method": "POST"
    }
},
"recurrence": {
    "frequency": "Week",
    "interval": 1,
    "schedule": {
        "weekDays": ["Monday", "Wednesday", "Friday"],
        "hours": [10, 22]
    },
    "count": 10,
    "endTime": "2012-11-04"
},
"state": "Disabled",
"status": {
    "lastExecutionTime": "2007-03-01T13:00:00Z",
    "nextExecutionTime": "2007-03-01T14:00:00Z",
    "executionCount": 3,
    "failureCount": 0,
    "faultedCount": 0
}
}
}

```

## startTime

In the startTime object, you can specify the start time and a time zone offset in ISO 8601 format.

## action

Your Scheduler job runs a primary action based on the specified schedule. Scheduler supports HTTP, Storage queue, Service Bus queue, and Service Bus topic actions. If the primary

action fails, Scheduler can run a secondary errorAction that handles the error. The action object describes these elements:

- The action's service type
- The action's details
- An alternative errorAction

The previous example describes an HTTP action. Here's an example for a Storage queue action:

```
"action": {  
    "type": "storageQueue",  
    "queueMessage": {  
        "storageAccount": "myStorageAccount",  
        "queueName": "myqueue",  
        "sasToken": "TOKEN",  
        "message": "My message body"  
    }  
}
```

Here's an example for a Service Bus queue action:

```
"action": {  
    "type": "serviceBusQueue",  
    "serviceBusQueueMessage": {  
        "queueName": "q1",  
        "namespace": "mySBNamespace",  
        "transportType": "netMessaging", // Either netMessaging or AMQP  
        "authentication": {  
            "sasKeyName": "QPolicy",  
            "type": "sharedAccessKey"  
        },  
        "message": "Some message",  
        "brokeredMessageProperties": {},  
        "customMessageProperties": {  
            "appname": "FromScheduler"  
        }  
    }  
},
```

Here's an example for a Service Bus topic action:

```
"action": {  
    "type": "serviceBusTopic",  
    "serviceBusTopicMessage": {  
        "topicPath": "t1",  
        "namespace": "mySBNamespace",  
        "transportType": "netMessaging", // Either netMessaging or AMQP  
        "authentication": {  
            "sasKeyName": "QPolicy",  
            "type": "sharedAccessKey"  
        }  
    }  
}
```

```
        "type": "sharedAccessKey"
    },
    "message": "Some message",
    "brokeredMessageProperties": {},
    "customMessageProperties": {
        "appname": "FromScheduler"
    }
}
},
```

## errorAction

If your job's primary action fails, Scheduler can run an errorAction that handles the error. In the primary action, you can specify an errorAction object so Scheduler can call an error-handling endpoint or send a user notification.

For example, if a disaster happens at the primary endpoint, you can use errorAction for calling a secondary endpoint, or for notifying an error handling endpoint.

Just like the primary action, you can have the error action use simple or composite logic based on other actions.

## recurrence

A job recurs if the job's JSON definition includes the recurrence object, for example:

```
"recurrence": {
    "frequency": "Week",
    "interval": 1,
    "schedule": {
        "hours": [10, 22],
        "minutes": [0, 30],
        "weekDays": ["Monday", "Wednesday", "Friday"]
    },
    "count": 10,
    "endTime": "2012-11-04"
},
```

Property	Required	Value	Description
frequency	Yes, when recurrence is used	"Minute", "Hour", "Day", "Week", "Month", "Year"	The time unit between occurrences
interval	No	1 to 1000 inclusively	A positive integer that determines the number of time units between each occurrence based on frequency
schedule	No	Varies	The details for more complex and advanced schedules. See hours, minutes, weekDays, months, and monthDays
hours	No	1 to 24	An array with the hour marks for when to run the job
minutes	No	0 to 59	An array with the minute marks for when to run the job
months	No	1 to 12	An array with the months for when to run the job
monthDay s	No	Varies	An array with the days of the month for when to run the job

---

weekDays	No	"Monday", "Tuesday", "Wednesday", "Thursday", "Friday", "Saturday", "Sunday"	An array with days of the week for when to run the job
count	No	<none>	The number of recurrences. The default is to recur infinitely. You can't use both count and endTime, but the rule that finishes first is honored.
endTime	No	<none>	The date and time for when to stop the recurrence. The default is to recur infinitely. You can't use both count and endTime, but the rule that finishes first is honored.

---

## retryPolicy

For the case when a Scheduler job might fail, you can set up a retry policy, which determines whether and how Scheduler retries the action. By default, Scheduler retries the job four more times at 30-second intervals. You can make this policy more or less aggressive, for example, this policy retries an action two times per day:

```
"retryPolicy": {  
  "retryType": "Fixed",  
  "retryInterval": "PT1D",  
  "retryCount": 2  
},
```

Property	Required	Value	Description
retryType	Yes	Fixed, None	Determines whether you specify a retry policy (fixed) or not (none).
retryInterval	No	PT30S	Specifies the interval and frequency between retry attempts in <a href="#">ISO 8601 format</a> . The minimum value is 15 seconds, while the maximum value is 18 months.
retryCount	No	4	Specifies the number of retry attempts. The maximum value is 20.

## state

A job's state is either Enabled, Disabled, Completed, or Faulted, for example:

```
"state": "Disabled"
```

To change jobs to Enabled or Disabled state, you can use the PUT or PATCH operation on those jobs. However, if a job has Completed or Faulted state, you can't update the state, although you can perform the DELETE operation on the job. Scheduler deletes completed and faulted jobs after 60 days.

## status

After a job starts, Scheduler returns information about the job's status through the status object, which only Scheduler controls. However, you can find the status object inside the job object. Here's the information that a job's status includes:

- Time for the previous execution, if any
- Time for the next scheduled execution for jobs in progress
- The number of job executions
- The number of failures, if any
- The number of faults, if any

For example:

```
"status": {
```

```
"lastExecutionTime": "2007-03-01T13:00:00Z",  
"nextExecutionTime": "2007-03-01T14:00:00Z ",  
"executionCount": 3,  
"failureCount": 0,  
"faultedCount": 0  
}
```

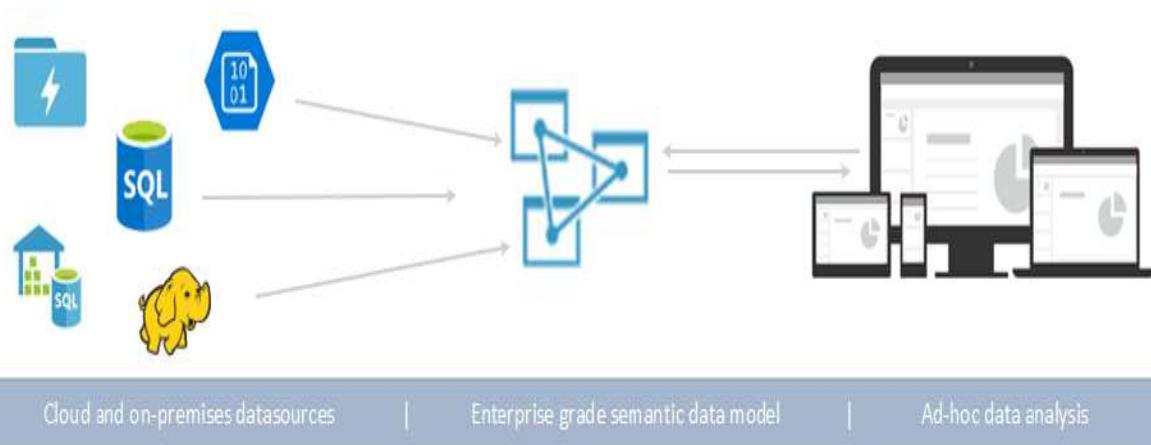
# AZURE ANALYSIS SERVICES



Azure Analysis Services

## What is Azure Analysis Services?

Azure Analysis Services is a fully managed platform as a service (PaaS) that provides enterprise-grade data models in the cloud. Use advanced mashup and modeling features to combine data from multiple data sources, define metrics, and secure your data in a single, trusted tabular semantic data model. The data model provides an easier and faster way for users to perform ad hoc data analysis using tools like Power BI and Excel.



## Get up and running quickly

In Azure portal, you can [create a server](#) within minutes. And with Azure Resource Manager [templates](#) and PowerShell, you can create servers using a declarative template. With a single template, you can deploy server resources along with other Azure components such as storage accounts and Azure Functions.

Azure Analysis Services integrates with many Azure services enabling you to build sophisticated analytics solutions. Integration with Azure Active Directory provides secure, role-based access to your critical data. Integrate with Azure Data Factory pipelines by including an activity that loads data into the model. Azure Automation and Azure Functions can be used for lightweight orchestration of models using custom code.

## The right tier when you need it

Azure Analysis Services is available in **Developer**, **Basic**, and **Standard** tiers. Within each tier, plan costs vary according to processing power, Query Processing Units (QPs), and memory size. When you create a server, you select a plan within a tier. You can change plans up or down within the same tier, or upgrade to a higher tier, but you can't downgrade from a higher tier to a lower tier.

### Developer tier

This tier is recommended for evaluation, development, and test scenarios. A single plan includes the same functionality of the standard tier, but is limited in processing power, QPs, and memory size. Query replica scale-out *is not available* for this tier. This tier does not offer an SLA.

DEVELOPER TIER		
Plan	QPUs	Memory (GB)
D1	20	3

### Basic tier

This tier is recommended for production solutions with smaller tabular models, limited user concurrency, and simple data refresh requirements. Query replica scale-out *is not available* for this tier. Perspectives, multiple partitions, and DirectQuery tabular model features *are not supported* in this tier.

BASIC TIER		
Plan	QPUs	Memory (GB)
B1	40	10

B2	80	16
----	----	----

## Standard tier

This tier is for mission-critical production applications that require elastic user-concurrency, and have rapidly growing data models. It supports advanced data refresh for near real-time data model updates, and supports all tabular modeling features.

<b>STANDARD TIER</b>		
<b>Plan</b>	<b>QPUs</b>	<b>Memory (GB)</b>
S0	40	10
S1	100	25
S2	200	50
S4	400	100
S8 <sup>1, 2</sup>	320	200
S9 <sup>1, 2</sup>	640	400
S8v2 <sup>1</sup>	640	200
S9v2 <sup>1</sup>	1280	400

1 - Not available in all regions.

2 - S8 and S9 are deprecated. v2 is recommended.

## Availability by region

Azure Analysis Services is supported in regions throughout the world. Supported plans and query replica availability depend on the region you choose. Plan and query replica availability can change depending on need and available resources for each region.

## Get started quickly and scale with efficiency

Use Azure Resource Manager to create and deploy an Azure Analysis Services instance within seconds and use backup restore to quickly move your existing models to Azure Analysis Services and take advantage of the scale, flexibility and management benefits of the cloud. Scale up, scale down or pause the service and pay only for what you use.



## Transform complex data into one version of the truth

Combine data from multiple sources into a single, trusted BI semantic model which is easy to understand and use. Enable self-service and data discovery for business users by simplifying the view of data and its underlying structure.



## Match performance to the speed of business

Reduce time-to-insights on large and complex datasets. Fast response times mean your BI solution can meet the needs of your business users and keep pace with your business. Connect to real-time operational data using DirectQuery and closely watch the pulse of your business.



## **Provide secured access anytime, from anywhere**

Make sure only authorised users can access your data models, no matter where they are, with role-based security and Azure Active Directory support. With 99.9% availability, your users can access critical information when they need it.



## **Accelerate time to delivery**

Release your BI solutions in a predictable and highly-secured way. Use the robust application lifecycle management capabilities to govern, deploy, test and deliver your BI solution quickly and with confidence.



## **Develop in a familiar environment**

Focus on solving business problems, not learning new skills, when you use the familiar, integrated development environment of Visual Studio. Easily deploy your existing SQL Server 2016 tabular models to the cloud.

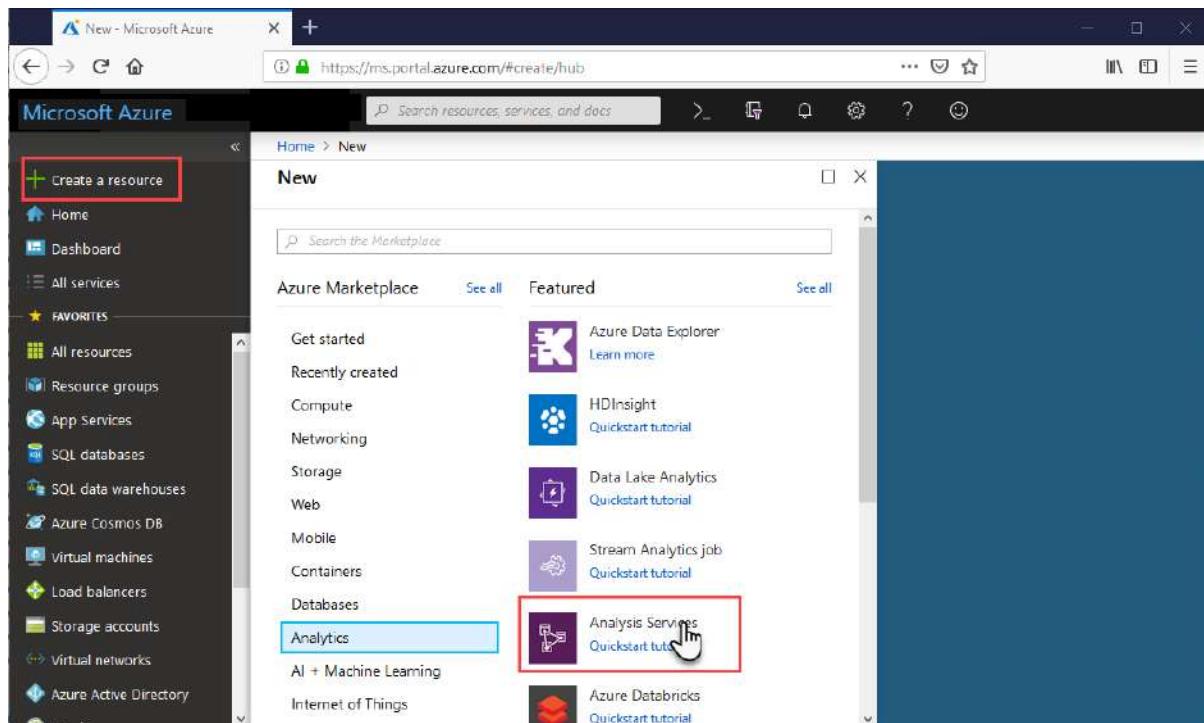


## Prerequisites

- **Azure subscription:** Visit Azure Free Trial to create an account.
- Azure Active Directory: Your subscription must be associated with an Azure Active Directory tenant. And, you need to be signed in to Azure with an account in that Azure Active Directory. Sign in to the Azure portal

## Create a server

1. Click **+ Create a resource > Analytics > Analysis Services.**



2. In **Analysis Services**, fill in the required fields, and then press **Create**.

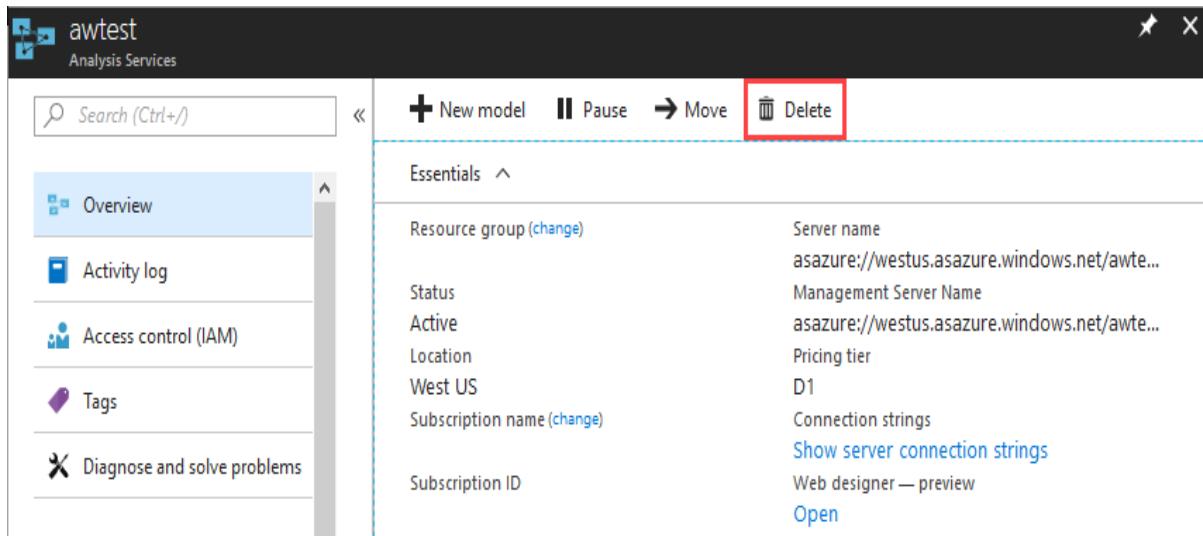
- **Server name:** Type a unique name used to reference the server. The server name must begin with a lowercase character and contain between 3 to 128 lowercase characters and numbers. Whitespace and special characters are not allowed.
- **Subscription:** Select the subscription this server will be associated with.

- **Resource group:** Create a new resource group or select one you already have. Resource groups are designed to help you manage a collection of Azure resources.
- **Location:** This Azure datacenter location hosts the server. Choose a location nearest your largest user base.
- **Pricing tier:** Select a pricing tier. If you are testing and intend to install the sample model database, select the free **D1** tier.
- **Administrator:** By default, this will be the account you are logged in with. You can choose a different account from your Azure Active Directory.
- **Backup Storage setting:** Optional. If you already have a you can specify it as the default for model database backup. You can also settings later.
- **Storage key expiration:** Optional. Specify a storage key expiration period.

Creating the server usually takes under a minute. If you selected **Add to Portal**, navigate to your portal to see your new server. Or, navigate to **All services > Analysis Services** to see if your server is ready. Servers support tabular models at the 1200 and higher compatibility levels. Model compatibility level is specified in Visual Studio or SSMS.

## Clean up resources

When no longer needed, delete your server. In your server's **Overview**, click **Delete**.

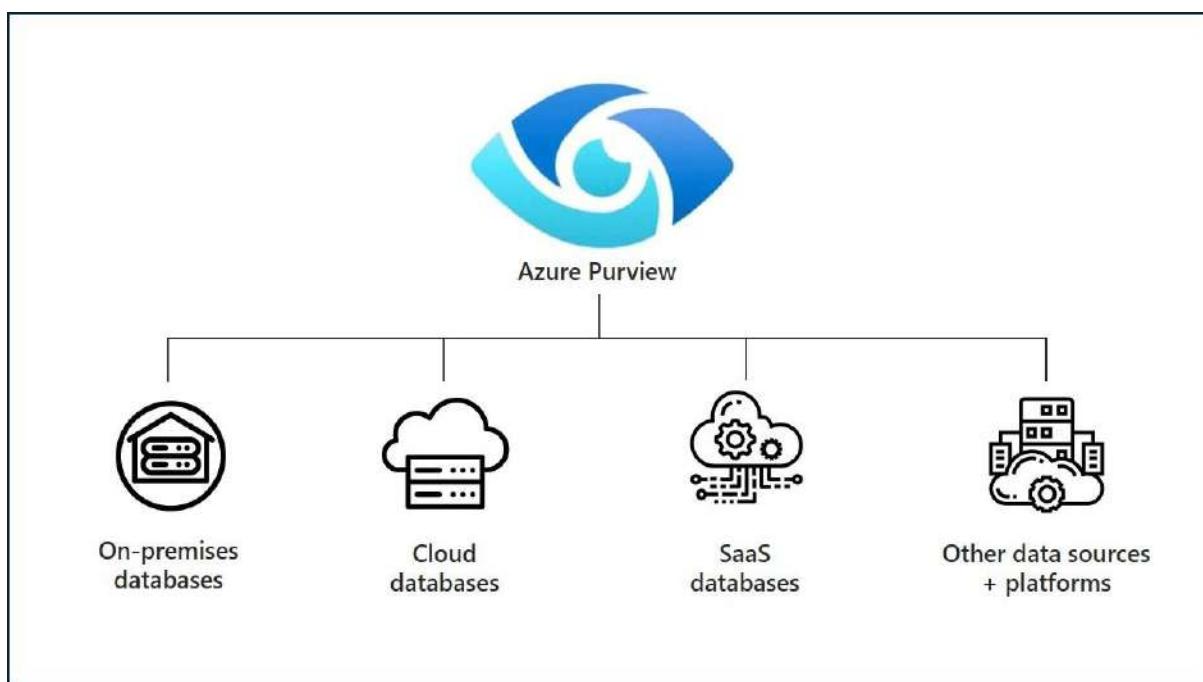


# AZURE DATA PURVIEW SERVICE



## What is “Azure Data Purview Service”?

Azure Purview is a unified data governance solution that helps you manage and govern your on-premises, multicloud, and software-as-a-service (SaaS) data. Easily create a holistic, up-to-date map of your data landscape with automated data discovery, sensitive data classification and end-to-end data lineage. Enable data consumers to find valuable, trustworthy data.



## **USES:**

**Automated data discovery, lineage identification, and data classification**



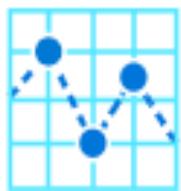
**Unified map of your data assets and their relationships for more effective data governance**



**Glossary with business and technical search terms to aid data discovery**



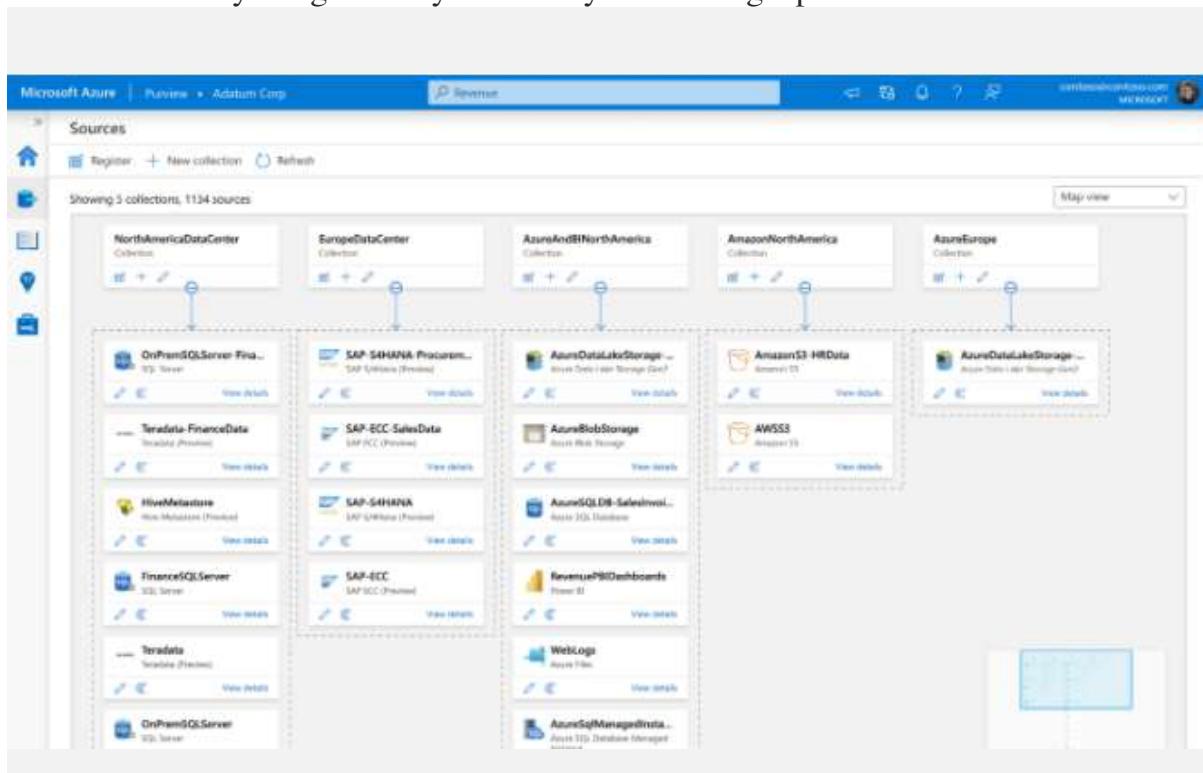
**Insights into the location and movement of sensitive data across your entire data estate**



## Create a unified map of your data across hybrid sources

Establish the foundation for effective data usage and governance with Purview Data Map.

- Automate and manage metadata from hybrid sources.
- Classify data using built-in and custom classifiers and Microsoft Information Protection sensitivity labels.
- Label sensitive data consistently across SQL Server, Azure, Microsoft 365, and Power BI.
- Easily integrate all your data systems using Apache Atlas APIs.

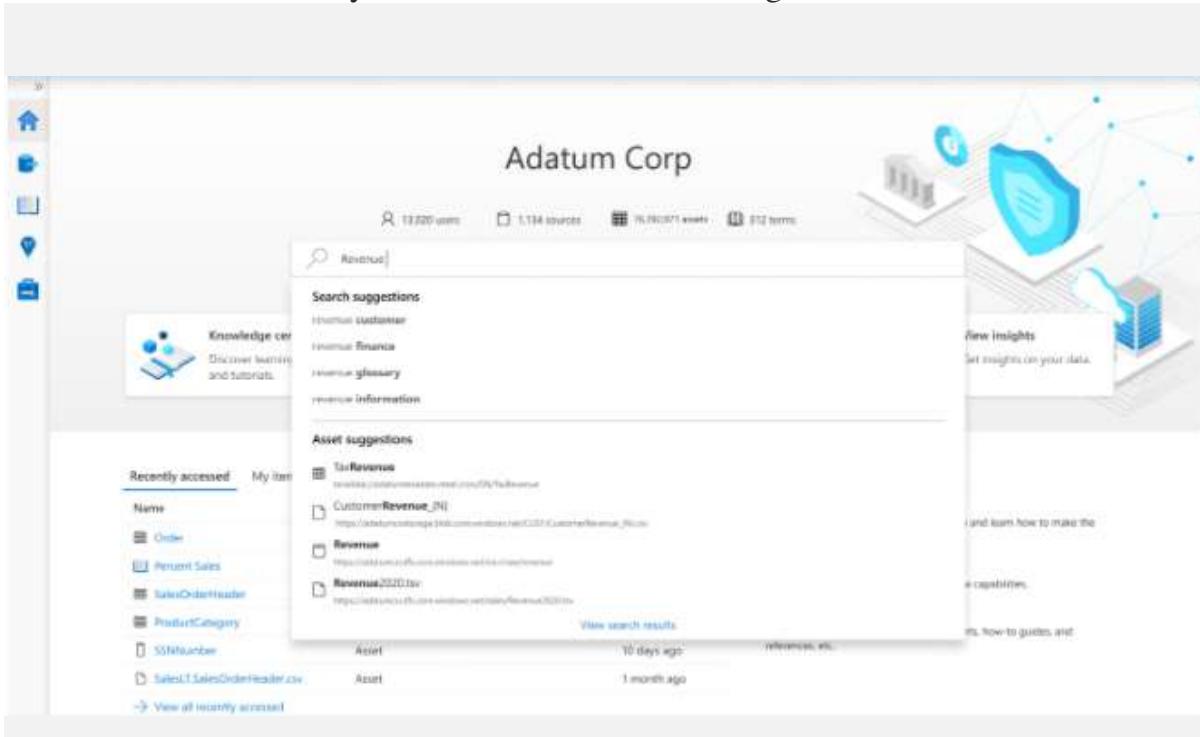


## Enable effortless discovery of trusted data

Maximize the business value of data for your data consumers with Purview Data Catalog.

- Search for data using technical or business terms.
- Easily understand data by browsing associated technical, business, semantic, and operational metadata.

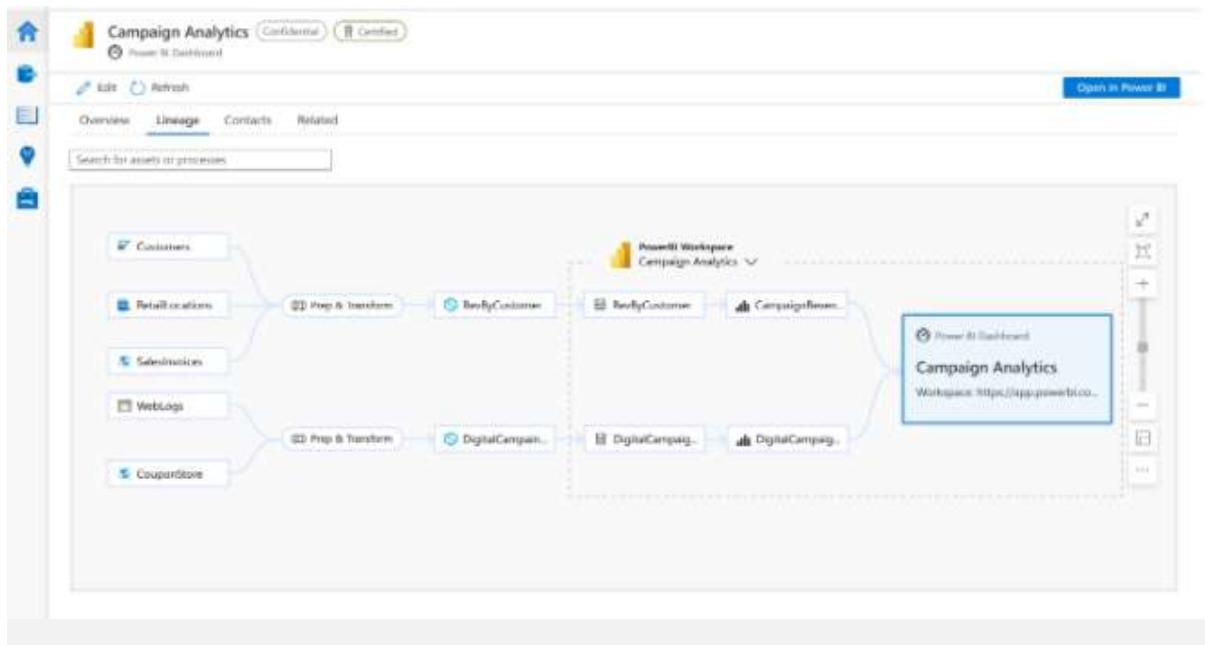
- Quickly identify the sensitivity level of data.
- Know where your data came from with interactive data lineage visualization.
- Empower data scientists, engineers, and analysts with business context to drive BI, analytics, AI, and machine learning initiatives.



## Discover data that powers business insights

Understand your data supply chain from raw data to business insights.

- Scan your Power BI environment and Azure Synapse Analytics workspaces with a few clicks and automatically publish all discovered assets and lineage to the Purview Data Map.
- Connect Azure Purview to Azure Data Factory instances to automatically collect data integration lineage.
- Quickly determine which analytics and reports already exist without reinventing the wheel.



## Create an Azure Purview account in the Azure portal

### Create an Azure Purview account

1. Go to the **Purview accounts** page in the .

Name	Type	Resource group	Location	Subscription	Status	Actions
ContosoPurview	Purview account	contosorg	East US	Contoso Subscription	Succeeded	... (More)

2. Select **Create** to create a new Azure Purview account.

The screenshot shows the Microsoft Azure Purview accounts page. At the top, there's a header bar with the Microsoft Azure logo and a search bar. Below it, the main title is 'Purview accounts'. A red box highlights the '+ Create' button in the top-left corner of the content area. Other buttons visible include 'View default account', 'Manage view', 'Refresh', 'Export to CSV', 'Open query', 'Assign tags', and 'Feedback'.

Or instead, you can go to the marketplace, search for **Azure Purview**, and select **Create**.

The screenshot shows the Microsoft Azure Marketplace search results for 'Azure Purview'. The search bar at the top has 'Azure Purview' entered. Below the search bar, there are filters for 'Pricing : All', 'Operating System : All', 'Publisher Type : All', and other options like 'Offer Type : All' and 'Publisher name : All'. The main area shows search results for 'Showing results for 'Azure Purview''. There are two items listed: 'Azure Purview' and 'Data Catalog'. Each item has a 'Create' button at the bottom. A red box highlights the 'Create' button for 'Azure Purview'. The left sidebar contains sections for 'Get Started', 'Service Providers', 'Management', 'Private Marketplace', 'My Marketplace', 'Favorites', 'Recently created', 'Private products', 'Categories', 'AI + Machine Learning', 'Analytics', 'Blockchain', 'Compute', 'Containers', and 'Databases'.

3. On the new Create Purview account page, under the **Basics** tab, select the Azure subscription where you want to create your Purview account.
4. Select an existing **resource group** or create a new one to hold your Purview account.

To learn more about resource groups, see our article on using resource groups to manage your Azure resources.

5. Enter a **Purview account name**. Spaces and symbols aren't allowed. The name of the Purview account must be globally unique. If you see the following error, change the name of Purview account and try creating again.

# Create Purview account

...

Provide Purview account info

Basics • Networking

Tags

Review + Create

Create a Purview account to develop a data governance solution in just a few clicks. A storage account and eventhub will be created in a managed resource group in your subscription for catalog ingestion scenarios. [Learn more](#)

## Project details

Subscription \*

< Your subscription selection >

Resource group \*

YourResourceGroup

▼

[Create new](#)

## Instance details

Purview account name \* ⓘ

TestPurview

✖ The name "TestPurview" is already in use. Please use a different name.

6. Choose a **location**. The list shows only locations that support Purview. The location you choose will be the region where your Purview account and meta data will be stored. Sources can be housed in other regions.
1. Select **Review & Create**, and then select **Create**. It takes a few minutes to complete the creation. The newly created Azure Purview account instance will appear in the list on your **Purview accounts** page.

**Create Purview account** ...

Provide Purview account info

\* Basics \* Networking Tags Review + Create

Create a Purview account to develop a data governance solution in just a few clicks. A storage account and eventhub will be created in a managed resource group in your subscription for catalog ingestion scenarios. [Learn more](#)

**Project details**

Subscription \* < Your subscription selection >

Resource group \* YourResourceGroup [Create new](#)

**Instance details**

Purview account name \* YourPurviewAccountName

Location \* East US

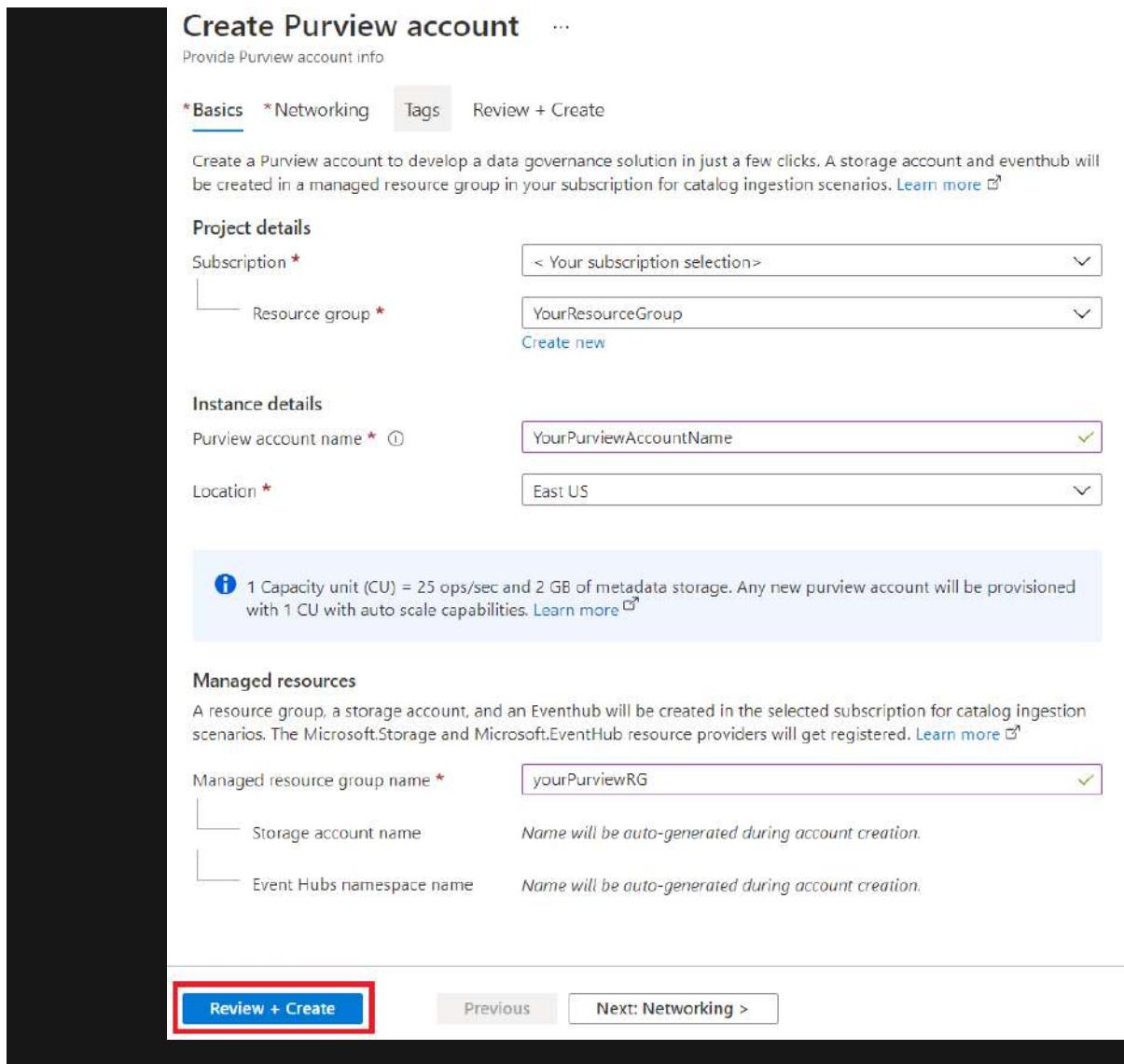
**Managed resources**

Managed resource group name \* yourPurviewRG

Storage account name Name will be auto-generated during account creation.

Event Hubs namespace name Name will be auto-generated during account creation.

**Review + Create** Previous Next: Networking >



## Open Purview Studio

After your Azure Purview account is created, you'll use the Purview Studio to access and manage it. There are two ways to open Purview Studio:

- Open your Purview account in the [Azure portal](#). Select the "Open Purview Studio" tile on the overview

p a g e .

# MyPurviewAccount

Purview account

Search (Ctrl+ /)

Refresh Delete

Overview

Activity log

Access control (IAM)

Tags

Diagnose and solve problems

Settings

Managed resources

Networking

Properties

Locks

Monitoring

Alerts

Metrics

Diagnostic settings

Automation

Tasks (preview)

Support + troubleshooting

Resource health

New Support Request

Essentials

Resource group  
contosoResourceGroup

Status  
Succeeded

Location  
East US

Subscription  
Contoso Subscription

Subscription ID  
abcdef01-2345-6789-0abc-def012345678

Tags (change)  
Click here to add tags

Type  
Purview account

Default account  
No

Platform size  
4 capacity units

Catalog  
C0 + C1

Data insights  
D0

Get Started

**Open Purview Studio**

Start using the unified data governance service and manage your hybrid data estate.

[Open](#)

**Manage users**

Grant users access to open Purview Studio.

[Go to Access control \(IAM\)](#)

**Documentation**

Learn how to be productive quickly. Explore concepts, tutorials, and other guidance available.

[Open](#)

This screenshot shows the Azure Purview Account overview page. The left sidebar contains navigation links for Overview, Activity log, Access control (IAM), Tags, Diagnose and solve problems, Settings, Managed resources, Networking, Properties, Locks, Monitoring, Alerts, Metrics, Diagnostic settings, Automation, Tasks (preview), Support + troubleshooting, Resource health, and New Support Request. The main content area displays account details under the 'Essentials' section, including Resource group (contosoResourceGroup), Status (Succeeded), Location (East US), Subscription (Contoso Subscription), Subscription ID (abcdef01-2345-6789-0abc-def012345678), and Tags. A red box highlights the 'Open Purview Studio' button in the 'Get Started' section, which is described as starting the unified data governance service. Other sections include 'Manage users' (Grant users access to open Purview Studio) and 'Documentation' (Learn how to be productive quickly). A large black redaction box covers the bottom portion of the page content.

# AZURE DATA CATALOG SERVICE



## What is Azure Data Catalog?

Azure Data Catalog is a [Data Catalog](#) cloud service of Microsoft using a crowdsourced approach. It provides an inventory of data used for discovering and understanding the data sources. Microsoft Azure is a Software as a Service ([SaaS](#)) application.

Azure Data Catalog enhances old investments' performance, adding metadata and notation around the Azure environment's data. It informs about the Data sources which we have discovered or which we already have.

It expresses documentation and describes the schema of the data source. The data source location and a copy of the metadata are present in the Azure Data Catalog. The user can access it easily when needed, and the indexing of metadata helps discover data through a search.

## Why do you need Azure Data catalog?

There were multiple challenges before employing Azure Data Catalog, be it for a consumer or a producer.

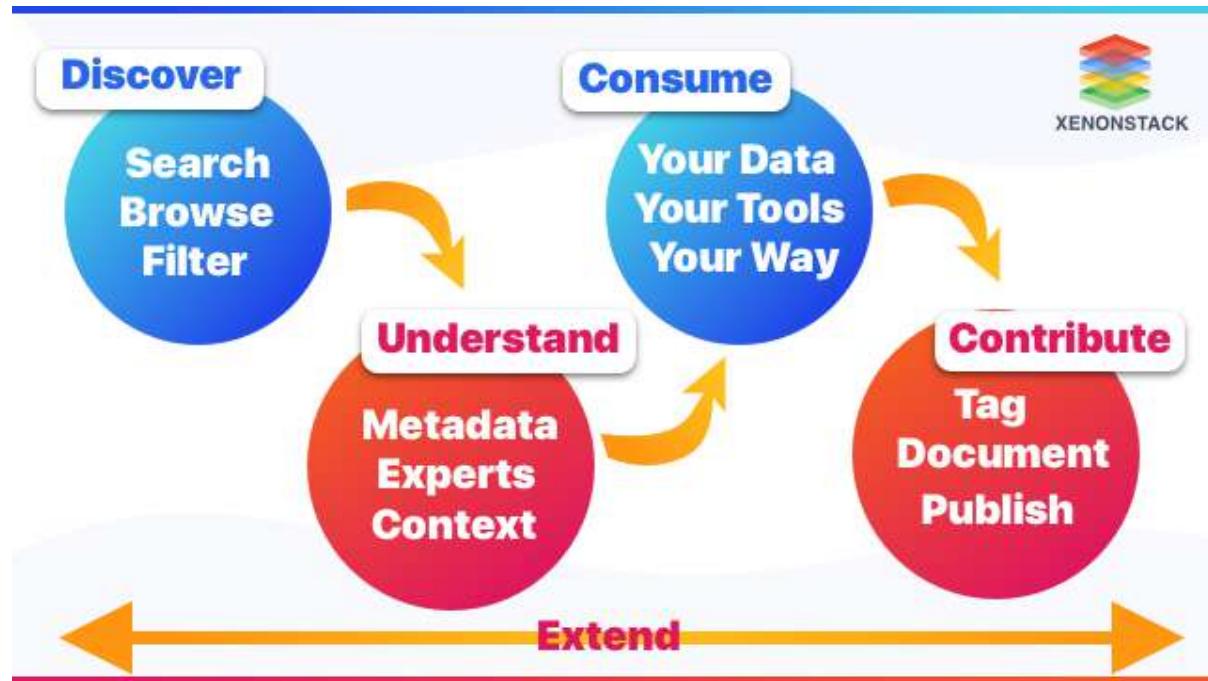
## **Challenges as a Consumer**

- Before Azure Data Catalog, the users were working only on the familiar datasets.
- Problems faced while selecting the data sources.
- Data availability issue.
- Without the information of the data, the consumer cannot connect to the data.
- No central location for data storage.
- Discovering data sources was based on tribal knowledge.
- Without documentation, a problem in understanding the data as well as its use.
- For queries related to the dataset, the consumer has to locate the producer responsible for the creation and ask them.
- Even after discovering and going through the documentation won't help, the consumer does not know how to request access to the data source.
- Time wasted in understanding the data, which can rather be used for the analysis.

## **Challenges as a Producer**

- Creating and maintaining the documentation for the data source is time-consuming and complex.
- Making documentation readily available to all the clients is also a challenge.
- As the data sources keep updating, there is a need to regularly update the documentation, which is an ongoing responsibility for the data producer.
- Annotating the source with metadata also won't help as the clients mostly ignore the data sources' descriptions.

# Azure Data Catalog Roadmap



Given below is a properly sequenced roadmap while working in Azure Data Catalogs:

1. Create a data catalog.
2. Go to Azure Portal.
3. Create a resource.
4. Select Data Catalog.
5. Name the Catalog and selecting the location for the catalog, and then creating it.
6. Publish the data
7. Go to the Data Catalog home page.
8. Click on Publish Data

Get to know the steps in **Azure Data Catalog Edition**:

1. Go to Settings.
2. Select the Catalog Edition. (Free or Standard)
3. Add Users
4. Go to Catalog Users
5. Click on Add Users
6. Portal Title (Expand Portal Title first, and then add text to be displayed on the Portal.)
7. Publish Page (Go to the Settings page, and then navigate to the Publish page and click on it.)

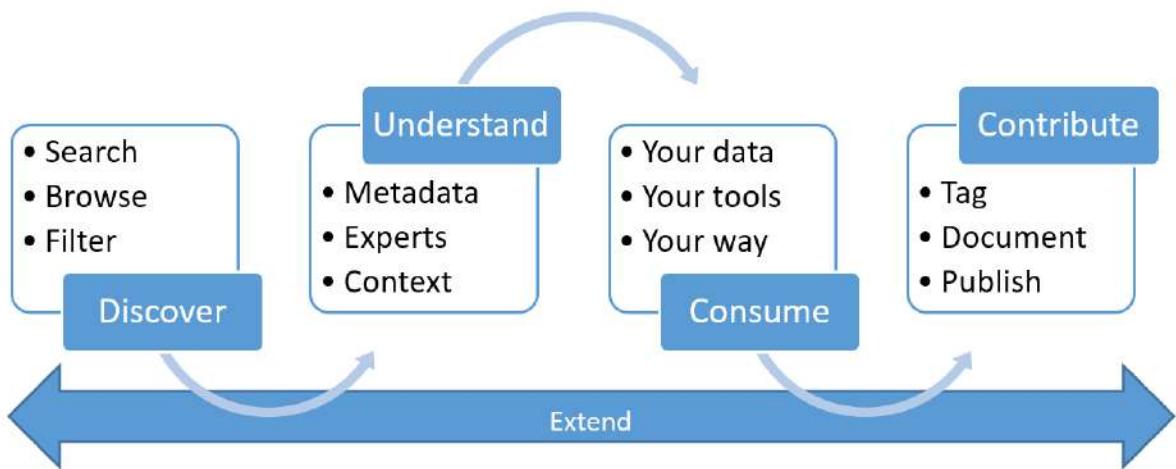
What about the steps involved in **Searching Data Catalog on Azure Portal**:

1. Navigate to Azure Portal and Sign In to the Account.
2. Go to All Services.
3. Select Data Catalog
4. Now you can see the Data Catalog which you just created.

## What is the Data Sources of Azure Data Catalog?

The metadata can be published using the public [API](#) or manually entering information directly into the Azure Data Catalog. Azure Data Catalog supports most of the data sources object that are supported are [Azure Data Lake](#) Store directory, Azure Data Lake Store file, Azure Blob storage, Azure Storage directory, [HDFS](#) directory, and file, Hive table, and view, MySQL table, and view, Oracle Database view and table, Teradata table and view, SAP Business Warehouse, SAP HANA view, Cassandra table and view, MongoDB table and view, etc.

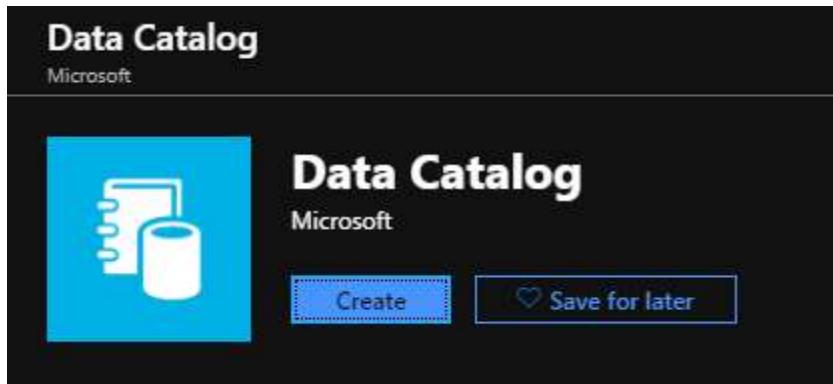




## Create an Azure Data Catalog via the Azure portal

Create a data catalog

1. Go to the [Azure portal](#) > **Create a resource** and select **Data Catalog**.



2. Specify a **name** for the data catalog, the **subscription** you want to use, the **location** for the catalog, and the **pricing tier**. Then select **Create**.
3. Go to the [Azure Data Catalog home page](#) and click **Publish Data**.



You can also get to the Data Catalog home page from the [Data Catalog service page](#) by selecting **Get started**.

The screenshot shows the Microsoft Azure Data Catalog landing page. At the top, there's a navigation bar with links like Overview, Solutions, Products (which is currently selected), Documentation, Pricing, Training, Marketplace, Partners, Support, Blog, More, Contact Sales, Search, My account, Portal, and Sign in. A 'Free account' button is also visible. Below the navigation is a large banner with a photo of two people looking at a tablet. The banner text reads 'Data Catalog' and 'Get more value from your enterprise data assets'. To the left is a list of benefits with checkmarks: 'Spend less time looking for data, and more time getting value from it', 'Register enterprise data assets', 'Discover data assets and unlock their potential', 'Capture tribal knowledge to make data more understandable', 'Bridge the gap between IT and the business, allowing everyone to contribute their insights', 'Let your data live where you want it, connect with the tools you choose', 'Control who can discover registered data assets', and 'Integrate into existing tools and processes with open REST APIs'. Below the banner is a 'Get started' button and some footer links: Explore Data Catalog, Pricing details, and Documentation.

4. Go to the **Settings** page.

The screenshot shows the 'Settings for your Azure Data Catalog' page. The header features a blue background with clouds and gears, and the text 'Settings for your Azure Data Catalog'. Below the header, it says 'Data Catalog administrators can update Catalog settings at any time.' The main section is titled 'Catalog Settings:' and includes fields for 'Data Catalog Name' (catalog), 'Subscription' (a blurred option), and 'Catalog Location' (westus). A note states: 'All metadata for data sources registered with Azure Data Catalog will be stored in the selected location.' Below this are several expandable sections: 'Pricing' (Free Edition - Unlimited users. Up to 5,000 registered data assets.), 'Security Groups' (Available with Standard Edition only.), 'Catalog Users' (1 Users Added - a blurred list), 'Glossary Administrators' (Available with Standard Edition only.), 'Catalog Administrators' (1 Users Added - a blurred list), 'Portal Title' (a blurred field), and 'Downgrade Catalog' (a blurred field). At the bottom right are 'Save' and 'Cancel' buttons.

5. Expand **Pricing** and verify your Azure Data Catalog **edition** (Free or Standard).

▼ **Pricing:** Free Edition - Up to 50 users. Up to 5,000 registered data assets.

The selected edition determines the number of users supported and the number of data assets that can be registered.

FREE EDITION	STANDARD EDITION
<input checked="" type="checkbox"/> Up to 50 users Up to 5,000 registered data assets	<input type="checkbox"/> Unlimited users, includes authorization Up to 100,000 registered data assets

6. If you choose **Standard** edition as your pricing tier, you can expand **Security Groups** and enable authorizing Active Directory security groups to access Data Catalog and enable automatic adjustment of billing.

**Security Groups:**

Enable authorizing Active Directory security groups to access Data Catalog and enable automatic adjustment of billing

**IMPORTANT:** By enabling the use of security groups, the actual list of users authorized to access Data Catalog is determined based on the users that are present in the authorized groups at any given time. If this option is selected, the number of Data Catalog billing users will be adjusted based on the number of users who are members of the authorized security groups. This may in turn affect the amount that you are billed for Data Catalog. If you do not want Data Catalog to automatically adjust the billing, please do not select this option. Instead, please authorize individual user accounts.

7. Expand **Catalog Users** and click **Add** to add users for the data catalog. You're automatically added to this group.

▼ **Catalog Users:** 1 Users Added -

Enable authorizing Active Directory security groups to access Data Catalog and enable automatic adjustment of billing

**IMPORTANT:** By enabling the use of security groups, the actual list of users authorized to access Data Catalog is determined based on the users that are present in the authorized groups at any given time. If this option is selected, the number of Data Catalog billing users will be adjusted based on the number of users who are members of the authorized security groups. This may in turn affect the amount that you are billed for Data Catalog. If you do not want Data Catalog to automatically adjust the billing, please do not select this option. Instead, please authorize individual user accounts.

Users that can register, annotate, discover, and use data sources.

**Add..**

8. If you choose **Standard** edition as your pricing tier, you can expand **Glossary Administrators** and click **Add** to add glossary administrator users. You're automatically added to this group.

## ▼ Glossary Administrators: 1 Users Added -

Administrators that can create, edit and delete terms within the Business Glossary.

**IMPORTANT:** Use of security groups is currently disabled.

**Add...**

9. Expand **Catalog Administrators** and click **Add** to add additional administrators for the data catalog. You're automatically added to this group.

## ▼ Catalog Administrators: 1 Users Added -

Administrators that can manage Catalog Settings, and can add and remove users. Note that registration of groups is not yet supported.

**Add...**

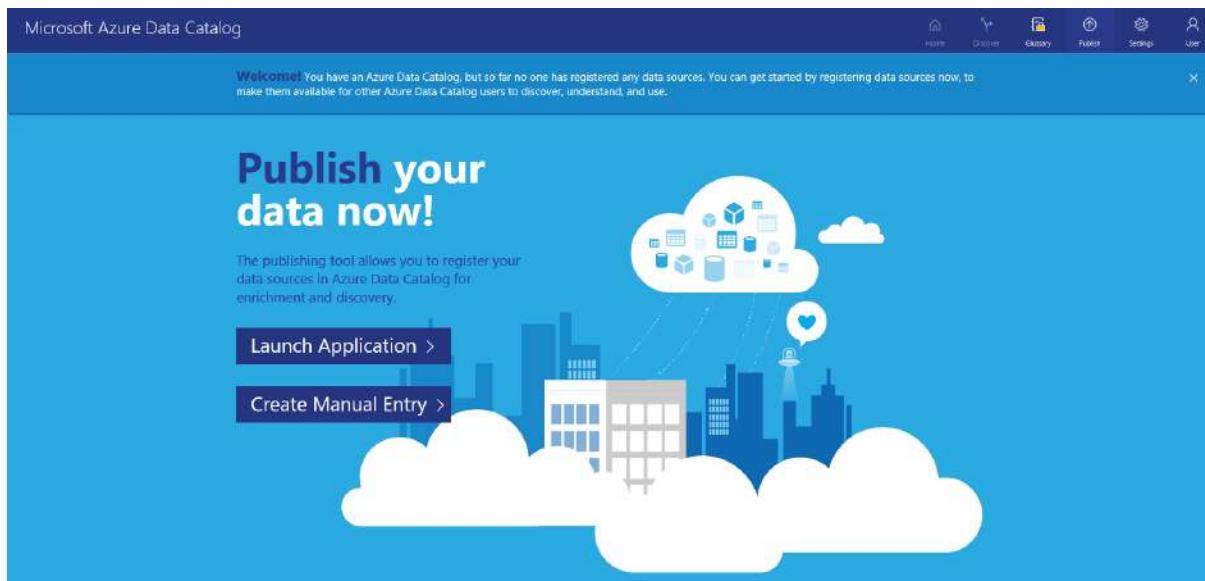
10. Expand **Portal Title** and add additional text that will be displayed in the portal title.

## ▼ Portal Title

Optional text that will be displayed in the Azure Data Catalog web portal title.

**IMPORTANT:** This text may not be displayed on displays with low resolution settings.

11. Once you complete the **Settings** page, next navigate to the **Publish** page.



Find a data catalog in the Azure portal

1. On a separate tab in the web browser or in a separate web browser window, go to the [Azure portal](#) and sign in with the same account that you used to create the data catalog in the previous step.
2. Select All services and then click **Data Catalog**.

The screenshot shows the Azure portal's "All services" blade. On the left, there's a sidebar with "Create a resource", "Home", "Dashboard", and a "Favorites" section containing "All resources", "Resource groups", "Analysis Services", "App registrations", "App Services", "Data Catalog" (which is highlighted), "Azure Active Directory", "Function Apps", and "SQL Database". The main area is titled "All services" with a search bar. It lists services under categories like "Everything", "General", "Compute", etc. A red box highlights the "Data Catalog" service under the "INTEGRATION" category.

You see the data catalog you created.

The screenshot shows the "Data Catalog" blade. At the top, it says "Data Catalog" and "Test\_Test\_PBIE\_ILDC\_VsabMsit1\_Subscription". There are buttons for "Edit columns", "Refresh", and "Assign tags". A message states "Only one Azure Data Catalog is supported per organization. If a Catalog has already been created for your organization, you cannot add additional catalogs." Below this, there's a table with a single item. The table has columns: NAME, TYPE, RESOURCE GROUP, and LOCATION. The single item is named "catalog", is a "Data Catalog", belongs to the "Test" resource group, and is located in "West US". The "NAME" column for this item has a red box around it.

3. Click the catalog that you created. You see the **Data Catalog** blade in the portal.

The screenshot shows the 'catalog' Data Catalog blade. At the top, there are 'Downgrade' and 'Move' buttons. Below them is a 'Essentials' section with dropdown menus for 'Resource group (change)', 'Location (West US)', and 'Subscription name (change)'. To the right of these are icons for cloud, users, and a clipboard. A 'Subscription ID' field is also present. At the bottom right is a 'All settings →' button.

**Pricing**

**Pricing tier**: catalog

**F 5K Up to 5,000 registered assets**  
Unlimited users

**Users**

4. You can view properties of the data catalog and update them. For example, click **Pricing tier** and change the edition.

The screenshot shows the same 'catalog' Data Catalog blade as before, but with a red box highlighting the 'Pricing tier' section. This section shows the current tier as 'catalog' with '5K Up to 5,000 registered assets' and 'Unlimited users'. To the right, a modal window titled 'Choose your pricing tier' is open, prompting the user to 'Please choose a pricing tier for your data catalog.' It lists two options: 'Standard' (100K assets, Unlimited users, AAD users and groups) and 'Free' (5K assets, Unlimited users, None Authorization). The 'Standard' option is selected and highlighted with a red border. The 'Standard' section shows a price of '1.00 USD/USER/MONTH (ESTIMATED)' and the 'Free' section shows '0.00 USD/MONTH'.

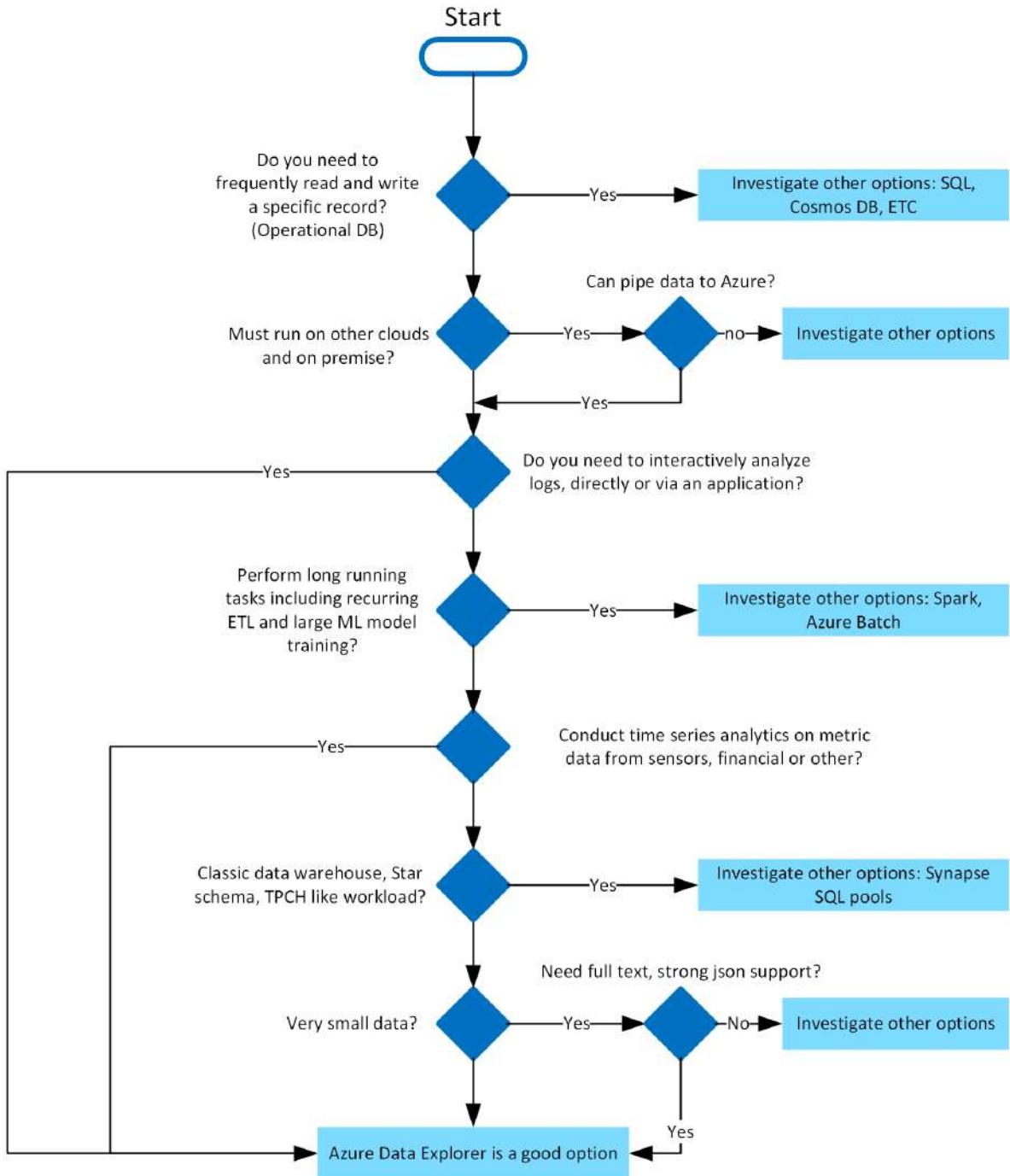
# AZURE DATA EXPLORER SERVICE

Azure Data Explorer is a fast, fully managed data analytics service for real-time analysis on large volumes of data streaming from applications, websites, IoT devices and more. Ask questions and iteratively explore data on the fly to improve products, enhance customer experiences, monitor devices and boost operations. Quickly identify patterns, anomalies and trends in your data. Explore new questions and get answers in minutes. Run as many queries as you need, thanks to the optimised cost structure.

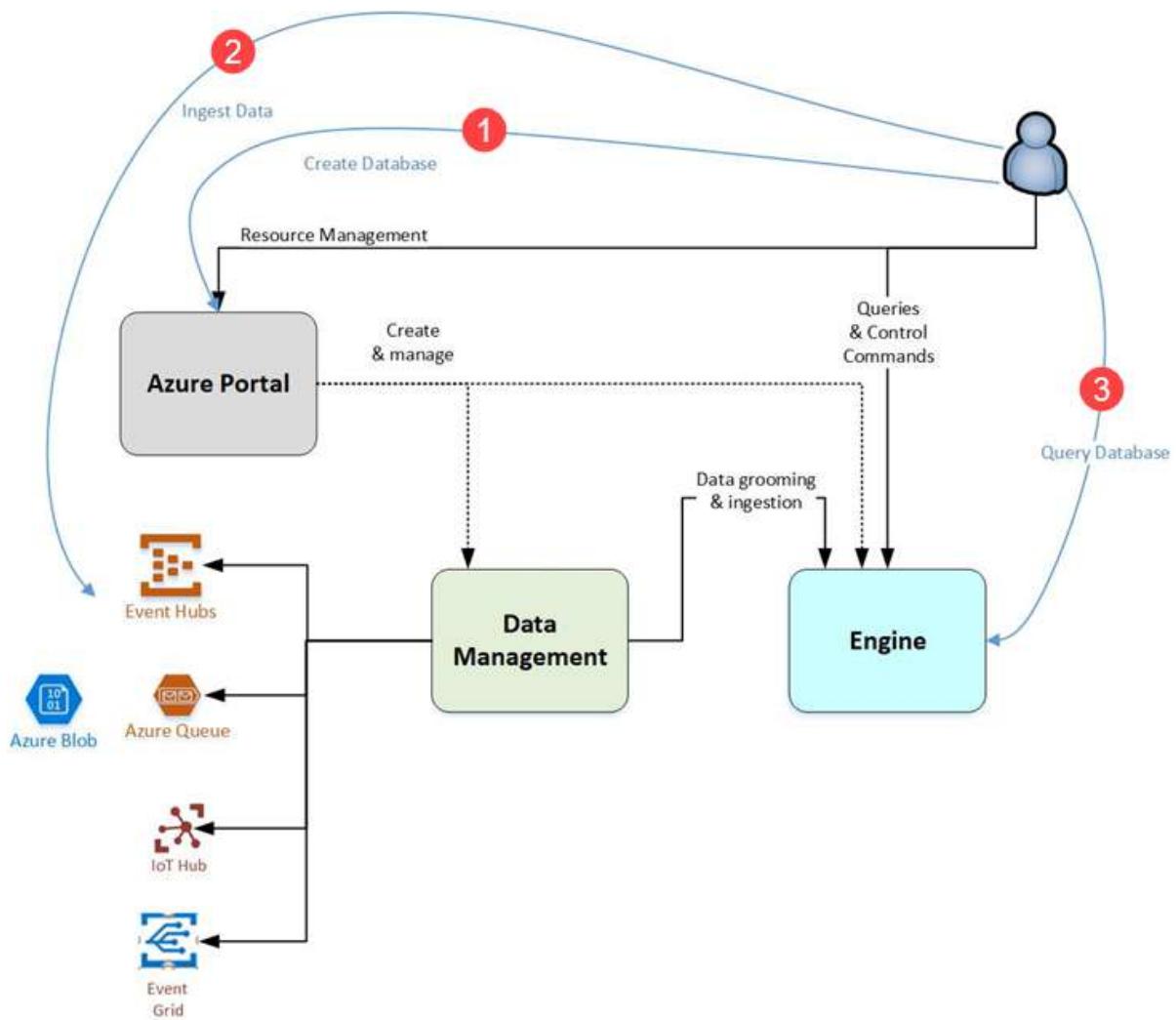
## What is Azure Data Explorer used for?

Azure Data Explorer is a fully managed, high-performance, big data analytics platform that makes it easy to analyze high volumes of data in near real time. The Azure Data Explorer toolbox gives you an end-to-end solution for data ingestion, query, visualization, and management.

By analyzing structured, semi-structured, and unstructured data across time series, and by using Machine Learning, Azure Data Explorer makes it simple to extract key insights, spot patterns and trends, and create forecasting models. Azure Data Explorer is scalable, secure, robust, and enterprise-ready, and is useful for log analytics, time series analytics, IoT, and general-purpose exploratory analytics.



## Azure Data Explorer flow



## Create an Azure Data Explorer cluster and database

Sign in to the Azure portal

Create a cluster

Create an Azure Data Explorer cluster with a defined set of compute and storage resources in an Azure resource group.

1. Select the **+ Create a resource** button in the upper-left corner of the portal.



#### Azure services



Create a  
resource



All resources



Azure Data  
Explorer...



Subscriptions



Shared  
dashboards



Azure Active  
Directory



SQL databases



Quickstart  
Center



Virtual  
machines



More services

#### Navigate



Subscriptions



Resource groups



All resources



Dashboard

#### Tools



Microsoft Learn  Learn Azure with free online training from Microsoft



Azure Monitor  Monitor your apps and infrastructure



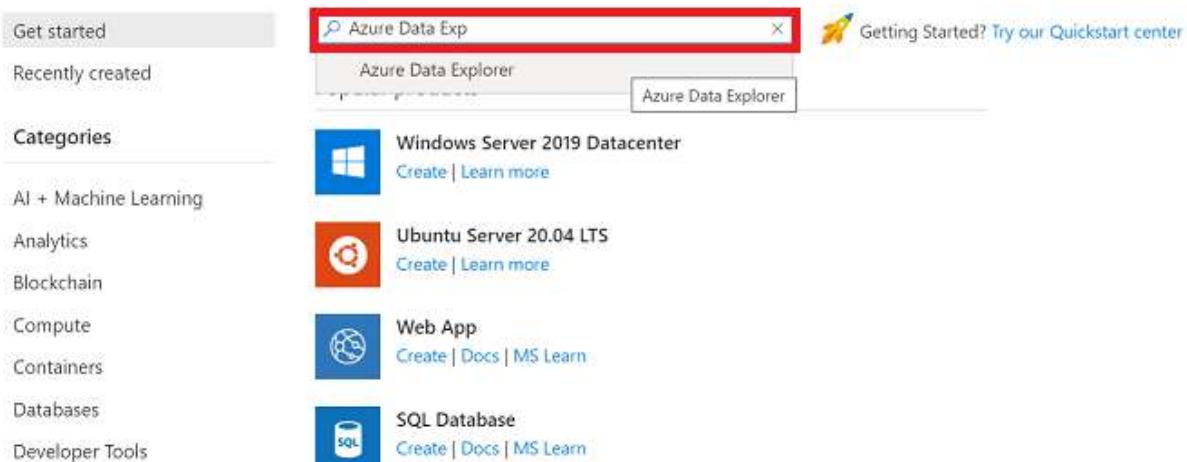
Security Center  Secure your apps and infrastructure



Cost Management  Analyze and optimize your cloud spend for free

## 2. Search for *Azure Data Explorer*.

### Create a resource



Get started

Recently created

Azure Data Explorer

Categories

- AI + Machine Learning
- Analytics
- Blockchain
- Compute
- Containers
- Databases
- Developer Tools

Azure Data Explorer

Windows Server 2019 Datacenter  

Ubuntu Server 20.04 LTS  

Web App   

SQL Database   

## 3. Under **Azure Data Explorer**, select **Create**.

### Azure Data Explorer

Microsoft



#### Azure Data Explorer

 Add to Favorites

Microsoft

★ 4.7 (31 Azure ratings)

Azure benefit eligible 

**Create**

[Overview](#) [Plans](#) [Usage Information + Support](#) [Reviews](#)

Azure Data Explorer is a big-data, interactive analytics platform that provides ultra-fast telemetry search and advanced text search for any type of data. Azure Data Explorer is perfect for IoT, troubleshooting and diagnostics, monitoring, security research, usage analytics, and more.

Azure Data Explorer is a modern, cloud-based, dynamically-scaling service, to meet all your business needs.

Azure Data Explorer makes it easy to optimize your total cost of ownership (TCO) - pay only for what you need, without worrying about upgrade and deployment costs and hassles.

4. Fill out the basic cluster details with the following information.

## Create an Azure Data Explorer Cluster ...

\* Basics   Scale   Configurations   Security   \* Network   Diagnostic settings   Tags   Review + create

**PROJECT DETAILS**

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription \*

Resource group \*  [Create new](#)

**CLUSTER DETAILS**

Cluster name \*

Region \*

Enable performance update (EngineV3)

**COMPUTE SPECIFICATION**

Workload \*

Size

Compute specifications \*  [Select from all options](#)

Availability zones

[Review + create](#) [Next : Scale >](#)

TABLE 1

Setting	Suggested value	Field description
Subscription	Your subscription	Select the Azure subscription that you want to use for your cluster.
Resource group	Your resource group	Use an existing resource group or create a new resource group.

Cluster name	A unique cluster name	Choose a unique name that identifies your cluster. The domain name <i>[region].kusto.windows.net</i> is appended to the cluster name you provide. The name can contain only lowercase letters and numbers. It must contain from 4 to 22 characters.
Region	West US or West US 2	Select <i>West US</i> or <i>West US 2</i> (if using availability zones) for this quickstart. For a production system, select the region that best meets your needs.
Workload	Dev/Test	Select <i>Dev/Test</i> for this quickstart. For a production system, select the specification that best meets your needs.
Compute specifications	Dev (No SLA)_Standard_E2a_v4	Select <i>Dev(No SLA)_Standard_E2a_v4</i> for this quickstart. For a production system, select the specification that best meets your needs.
Availability zones	1, 2, or 3	Place the cluster instances in one or more availability zones in the same region (optional). <a href="#">Azure Availability Zones</a> are unique physical locations within the same Azure region. They protect an Azure Data Explorer cluster from loss data. The cluster nodes are created, by default, in the same data center. When you select several availability zones you can eliminate a single point of failure and ensure high availability. <b>Deployment to availability zones is possible only when creating the cluster, and can't be modified later.</b>

5. Select **Review + create** to review your cluster details, and on the next screen select **Create** to provision the cluster. Provisioning typically takes about 10 minutes.
6. When the deployment is complete, select **Go to resource**.

**Microsoft.AzureKusto - Overview**

Deployment

Search (Ctrl+/  
)

Delete Cancel Redeploy Refresh

Overview Inputs Outputs Template

Your deployment is complete

Go to resource

Deployment name: Microsoft.Azure  
Subscription:

Resource group:

DEPLOYMENT DETAILS (Download)

Deployment name: clustertest  
Subscription:  
Resource group:  
Start time: 10/7/2021, 12:40:16 PM  
Correlation ID: 6ae25

RESOURCE	TYPE	STATUS	OPERATI...
✓	Microsoft...	OK	Operation

## Create a database

You're now ready for the second step in the process: database creation.

### 1. On the Overview tab, select **Create database**.

Search (Ctrl+/  
)

Add database Stop Refresh Move Delete Feedback

Overview Activity log Access control (IAM) Tags Diagnose and solve problems Permissions Query Scale up Scale out Configurations Identity Encryption Properties Locks Databases

To use Azure Data Explorer, create at least one database. →

Essentials

Resource group (change) : Location : West US 2 Subscription (change) : 32-6 Subscription ID : 220fc532-6 Engine type : V3

State : Running URI : Data Ingestion URI : Compute specifications : Dev[No SLA]\_Standard\_E2a\_v4 Instance count : 1

Welcome to Azure Data Explorer

STEP 1 - CLUSTER CREATION

STEP 2 - DATABASE CREATION

Your cluster was successfully created.  
Now you can create a database.

Create database

STEP 3 - DATA INGESTION

2. Fill out the form with the following information.

## Azure Data Explorer Database

Create new database

Admin ⓘ

Database name \*

Retention period (in days) ⓘ

Unlimited days for retention period

Cache period (in days) ⓘ

Unlimited days for cache period

**Create**

TABLE 2

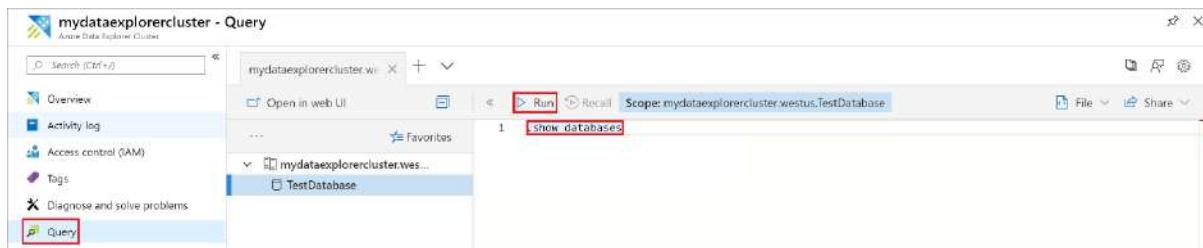
Setting	Suggested value	Field description
Admin	<i>Default selected</i>	The admin field is disabled. New admins can be added after database creation.
Database name	<i>TestDatabase</i>	The database name must be unique within the cluster.
Retention period	<i>365</i>	The time span (in days) for which it's guaranteed that the data is kept available to query. The time span is measured from the time that data is ingested.
Cache period	<i>31</i>	The time span (in days) for which to keep frequently queried data available in SSD storage or RAM, rather than in longer-term storage.

3. Select **Create** to create the database. Creation typically takes less than a minute.  
When the process is complete, you're back on the cluster **Overview** tab.

Run basic commands in the database

After you created the cluster and database, you can run queries and commands. The database doesn't have data yet, but you can still see how the tools work.

1. Under your cluster, select **Query**. Paste the command `.show databases` into the query window, then select **Run**.



The result set shows **TestDatabase**, the only database in the cluster.

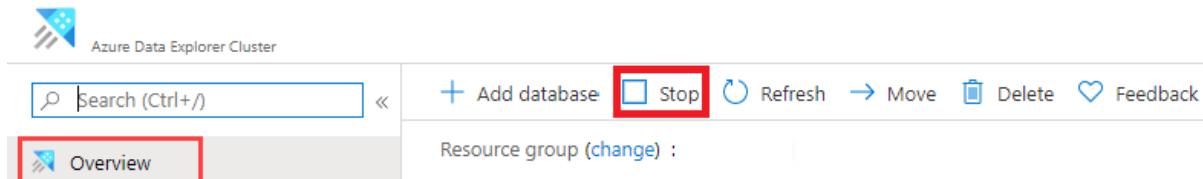
2. Paste the command `.show tables` into the query window and select **Run**.

This command returns an empty result set because you don't have any tables yet. You add a table in the next article in this series.

Stop and restart the cluster

You can stop and restart a cluster depending on business needs.

1. To stop the cluster, at the top of the **Overview** tab, select **Stop**.



1. To restart the cluster, at the top of the **Overview** tab, select **Start**.

When the cluster is restarted, it takes about 10 minutes for it to become available (like when it was originally provisioned). It takes more time for data to load into the hot cache.

Clean up resources

If you plan to follow other quickstarts and tutorials, keep the resources you created. Otherwise, clean up your resource group, to avoid incurring costs.

1. In the Azure portal, select **Resource groups** on the far left, and then select the resource group that contains your Data Explorer cluster.

2. Select **Delete resource group** to delete the entire resource group. If using an existing resource group, you can choose to only delete the Data Explorer cluster.

## Service Capabilities

### Low-latency ingestion

Elastically scale to terabytes of data in minutes. This data management service offers fast, low-latency ingestion with linear scaling which supports up to 200 MB of data per second per node. Azure Data Explorer supports a growing number of ingestion methods of data from devices, applications, servers and services for your specific use cases.



### Fast read-only query with high concurrency



Get results from 1 billion records in less than a second without modifying the data or metadata. Continue refining your queries until you have completed your analysis. Query large amounts of structured, semi-structured (JSON-like nested types) and unstructured (free-text) data. Search for specific text terms, locate events and perform calculations on structured data. The intuitive query language uses Microsoft IntelliSense options and colour coding to help you quickly spot patterns, trends and anomalies. Simplify data exploration with fast text indexing, column store and time-series operations all in one service.

## Time-series analysis

Create and analyse thousands of time series in seconds with near-real-time monitoring solutions and workflows. Azure Data Explorer includes native support for creation, manipulation and analysis of multiple time series.



## Fully managed data service

Focus on the data instead of the infrastructure. This powerful, fully managed data analytics service automatically scales to meet your demands. Control costs by paying only for what you need, with no upfront costs or termination fees. Take advantage of the global availability for massive scalability.



## Cost-effective queries and storage

Ask unlimited questions without skyrocketing costs; you pay by the hour, not by the query. You also control your storage costs. Get the best of a persistent database to automatically add data to the table, but with the flexibility to choose a retention policy based on how long you want to store the data. For persistent storage at commodity pricing, write data to Azure Blob Storage for future use.



## Custom solutions with built-in analytics

Using our platform as a service (PaaS), build your own solution with interactive analytics built in. Azure Data Explorer is the data service for Azure Monitor, Azure Time Series Insights and Windows Defender Advanced Threat Protection. It supports REST API, MS-TDS and Azure Resource Manager service endpoints and several client libraries.



# AZURE DATA FACTORY SERVICE

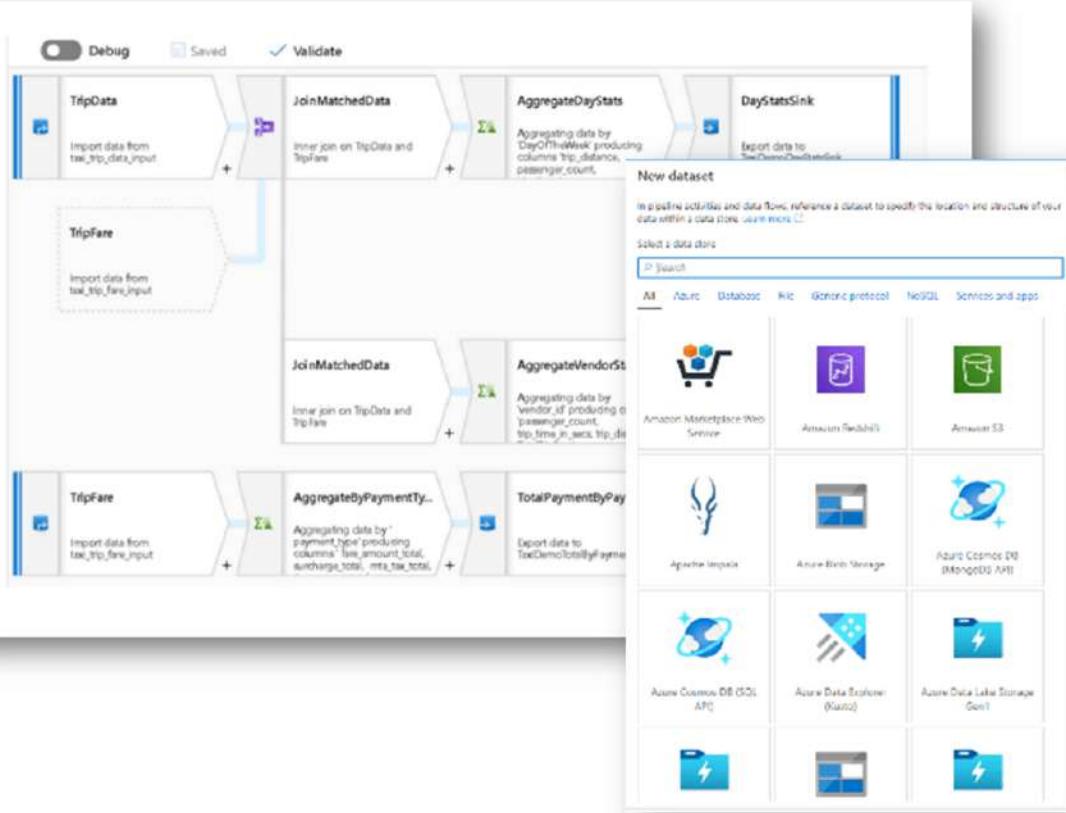


## WHAT IS AZURE DATA FACTORY?

Azure Data Factory is Azure's cloud ETL service for scale-out serverless data integration and data transformation. It offers a code-free UI for intuitive authoring and single-pane-of-glass monitoring and management.

You can also lift and shift existing SSIS packages to Azure and run them with full compatibility in ADF. SSIS Integration Runtime offers a fully managed service, so you don't have to worry about infrastructure management.

Azure Data Factory (ADF) is a fully managed, serverless data integration solution for ingesting, preparing, and transforming all your data at scale. It enables every organization in every industry to use it for a rich variety of use cases: data Engineering, migrating their on-premises SSIS packages to Azure, operational data integration, analytics, ingesting data into data warehouses, and more.



## What does Azure Data Factory do?

It allows you to:

- **Copy** data from many supported sources both on-premise and cloud sources
- **Transform** the data (cf. below paragraphs)
- **Publish** the copied and transformed data, sending it to a destination data storage or analytics engine
- **Monitor** the data flows using a rich graphical interface

## What doesn't Azure Data Factory do?

Data Factory isn't SSIS (SQL Server Integration Services) in the cloud. It has less database specific features and focuses on supporting broader data transformation & movements (incl. big datasets, incl. data lake operations).

Data Factory can, however, run your SSIS packages in the Cloud (once build in SSIS). This allows to leverage Data Factory's scalability with SSIS's advanced ETL features.

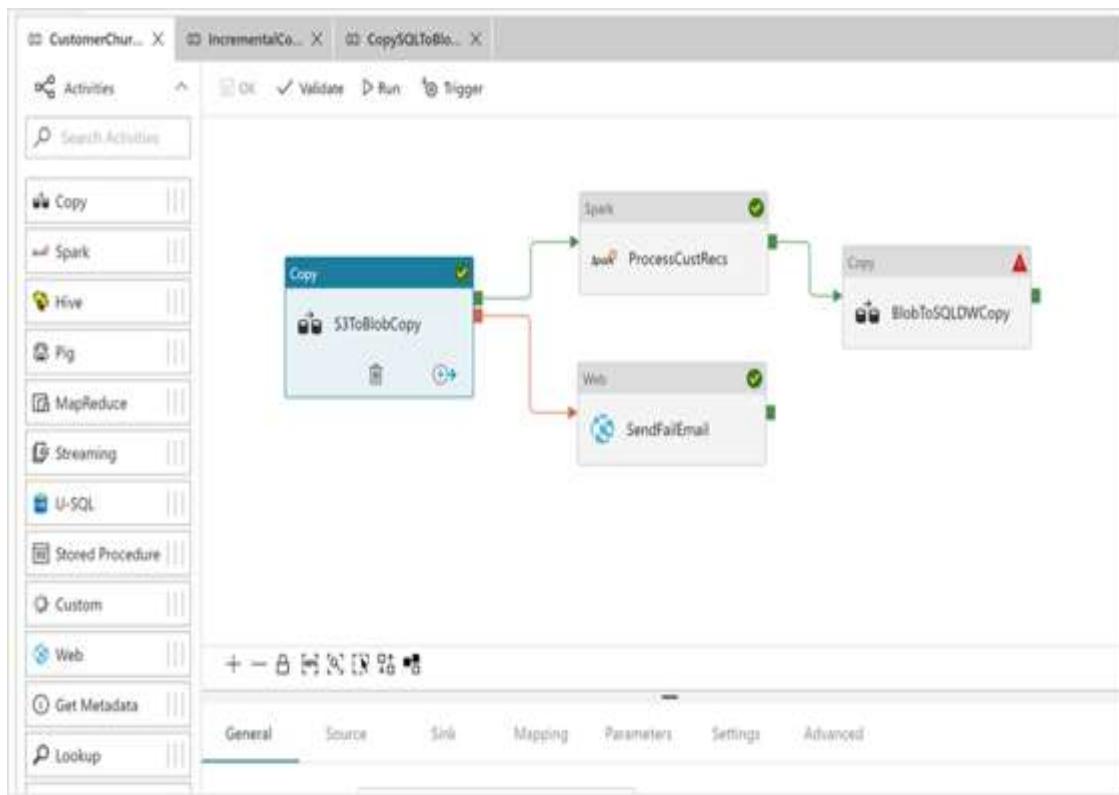
## Why do I need Azure Data Factory?

Data Factory is an enabler for any Cloud projects. In almost any Cloud project you will need to perform data movement activities across various networks (on-premise network and Cloud) and across various services (i.e. from and to close different Azure storages).

Data Factory is particularly a required enabler for organizations who are making their first steps in the Cloud & who thus try to connect on-premise data with the Cloud. For this Azure Data Factory has an Integration Runtime engine, a Gateway service which can be installed on-premise which guarantees performant & secure transfer of data from & to the cloud.

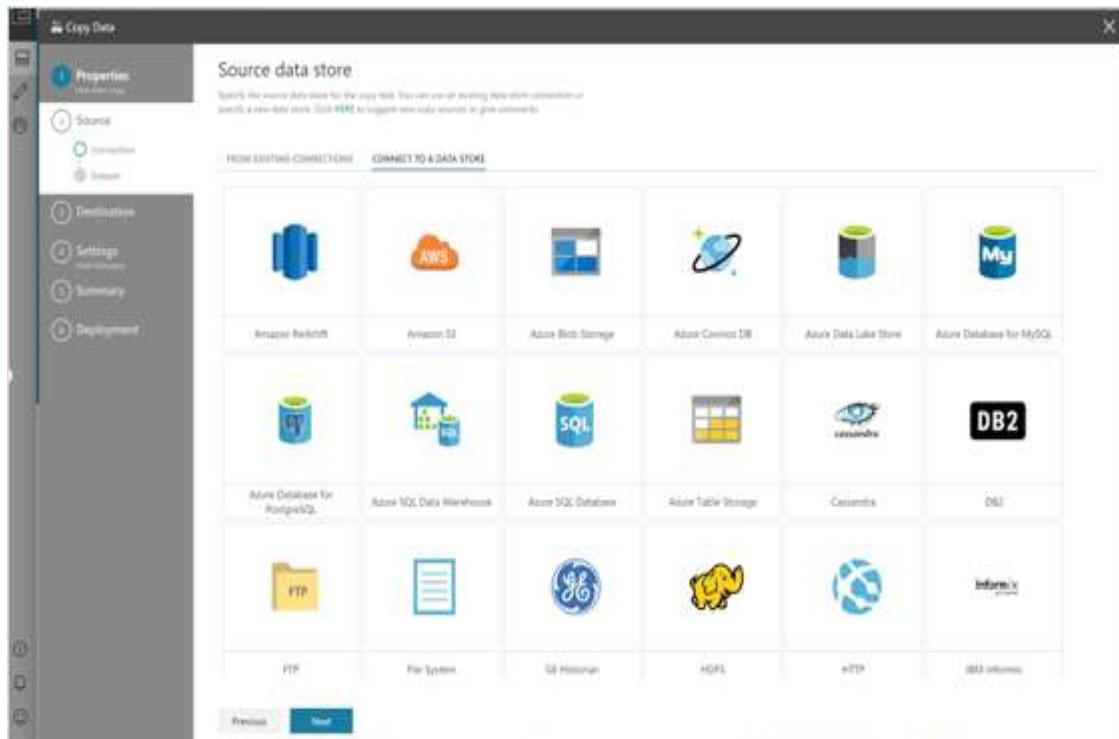
## How do I work with Azure Data Factory?

Azure Data Factory is a user interface tool which offers a very graphical overview to create/manage activities and pipelines. It doesn't require coding skills, yet complex transformation will require Azure Data Factory experience.



Important features:

- Azure Data Factory has default connectors with close to all on-premise data sources including MySQL, SQL Server, Oracle DBs



- Azure Data Factory supports **branching**, where the output of one activity can be a trigger for the start of another activity.  
- e.g. first copy the data from on-premise to Blob, then merge all blobs
- Azure Data Factory support **tumbling window trigger** & event trigger. The first is particularly relevant in creating partitioned data in for example a Data Lake set-up (for example storing your data automatically in daily partitioned blobs: e.g. YYYY/MM/DD/Blob.csv).  
An **event trigger** is applicable when an event such as a new Blob on Blob Storage should automatically trigger a transformation.
- Azure Data Factory allows to work with **parameters** and thus enables to pass on dynamically parameters between datasets, pipelines & triggers. An example could be that the filename of the destination file should have the name of the pipeline or should be the date of the data slice.
- Azure Data Factory allows to run pipeline **up to 1 run per minute**. It thus doesn't allow real-time but enables close to real-time.
- Azure Data Factory provides **monitoring & alerting**. The execution of the different pipelines can be easily monitored through the UI & you can set-up alerts (linked to Azure Monitor) if anything fails.

The screenshot shows the Microsoft Azure Pipeline Runs dashboard. At the top, there are tabs for Pipeline Runs, Integration Runtimes, Trigger Runs, and Alerts. Below the tabs are buttons for Run, Cancel, Refresh, Alerts, and Metrics. The main area displays a table of pipeline runs. A filter bar at the top of the table allows setting a Custom Range (from 10/16/2018 6:00 PM to 10/24/2018 6:00 PM) and a Time Zone (UTC+01:00: Amsterdam, Berlin, Bern, Rome, Stockholm, Vienna). The table has columns for Pipeline Name, Actions, Run Start, Duration, Triggered By, Status, and Parameter. One row is visible, showing 'pipeline1' with a status of 'Succeeded'.

Pipeline Name	Actions	Run Start	Duration	Triggered By	Status	Parameter
pipeline1		10/17/2018, 10:31:37 AM	00:06:00	Manual trigger	Succeeded	

- Azure Data Factory can work well with Azure Databricks to schedule ML algorithms.

# Should I use Azure Data Factory or SSIS?

Use the right tool for the right purpose. Through below overview you understand that they are complementary. They are also built that way: i.e., Azure Data Factory also offers the ability to deploy, manage and run SSIS packages in managed Azure SSIS Integration Runtimes. Based on your current platform/solution:

	Hybrid On-Prem & Azure Solution	Azure Solution	On-Prem Only Solution
Azure Data Factory (ADF V2)	Yes	Yes	No
Integration Services (SSIS)	Yes	Yes	Yes

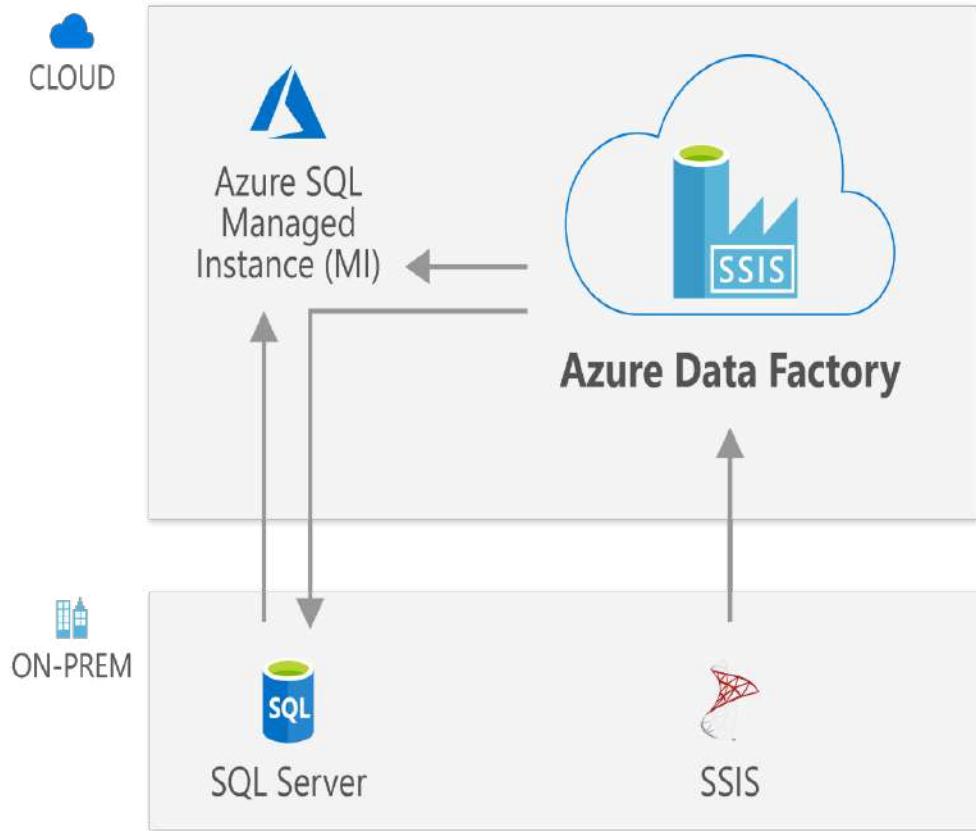
Based on type of data:

	Small data	Close to real-time data (every minute)	Big Data
Azure Data Factory (ADF V2)	Yes	Yes	Yes
Integration Services (SSIS)	Yes	No	No

## Rehost and extend SSIS in a few clicks

Azure Data Factory can help organizations looking to modernize SSIS.

- Realize up to 88 percent cost savings with the Azure Hybrid Benefit.
- Enjoy the only fully compatible service that makes it easy to move all your SSIS packages to the cloud.
- Migration is easy with the deployment wizard and ample how-to documentation.
- Realize your vision for hybrid big data and data warehousing initiatives by combining with Data Factory cloud data pipelines.



## Ingest all your data with built-in connectors

Ingesting data from diverse and multiple sources can be expensive, time consuming and require multiple solutions. Azure Data Factory offers a single, pay-as-you-go service. You can:

- Choose from more than 90 built-in connectors to acquire data from Big Data sources like Amazon Redshift, Google BigQuery, HDFS; enterprise data warehouses like Oracle Exadata, Teradata; SaaS apps like Salesforce, Marketo, and ServiceNow; and all Azure data services.
- Use the full capacity of underlying network bandwidth, up to 5 GB/s throughput.



## Hybrid data integration, simplified

In today's data-driven world, big data processing is a critical task for every organisation. To unlock transformational insights, data engineers need services that are built to simplify ETL as well as handle the complexities and scale challenges of big data integration.

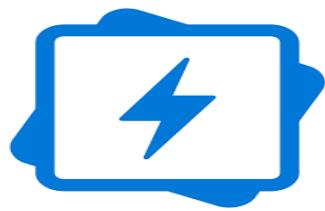
With Azure Data Factory, it is fast and easy to build code-free or code-centric ETL and ELT processes. In this scenario, learn how to create code-free pipelines within an intuitive visual environment.



Azure Data Factory trusted by companies of all sizes

- 1) Adobe
- 2) Concentra

# AZURE DATA LAKE ANALYTICS SERVICE



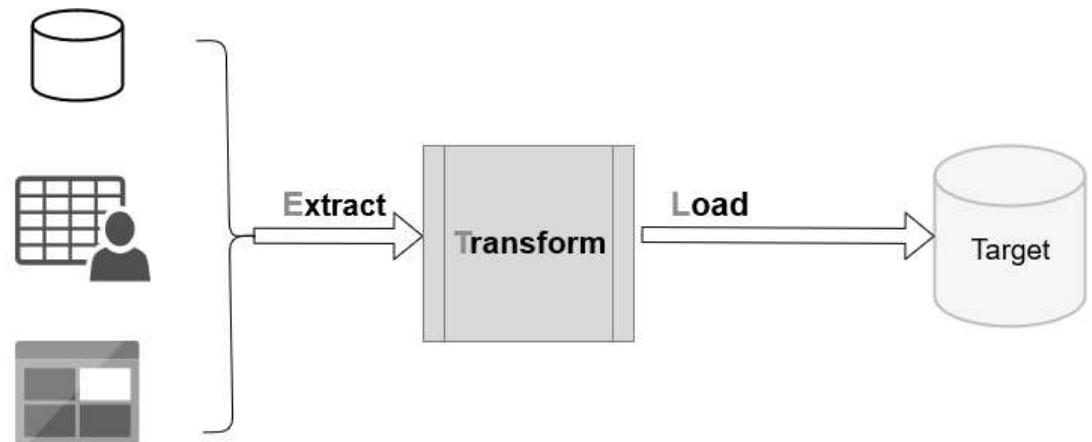
## WHAT IS AZURE DATA LAKE ANALYTICS?

Microsoft Azure platform supports big data such as Hadoop, HDInsight, Data lakes. Usually, a traditional data warehouse stores data from various data sources, transform data into a single format and analyze for decision making. Developers use complex queries that might take longer hours for data retrieval. Organizations are increasing their footprints in the Cloud infrastructure.

It leverages cloud infrastructure warehouse solutions such as Amazon RedShift, Azure Synapse Analytics (Azure SQL data warehouse), or AWS snowflake. The cloud solutions are highly scalable and reliable to support your data and query processing and storage requirements.

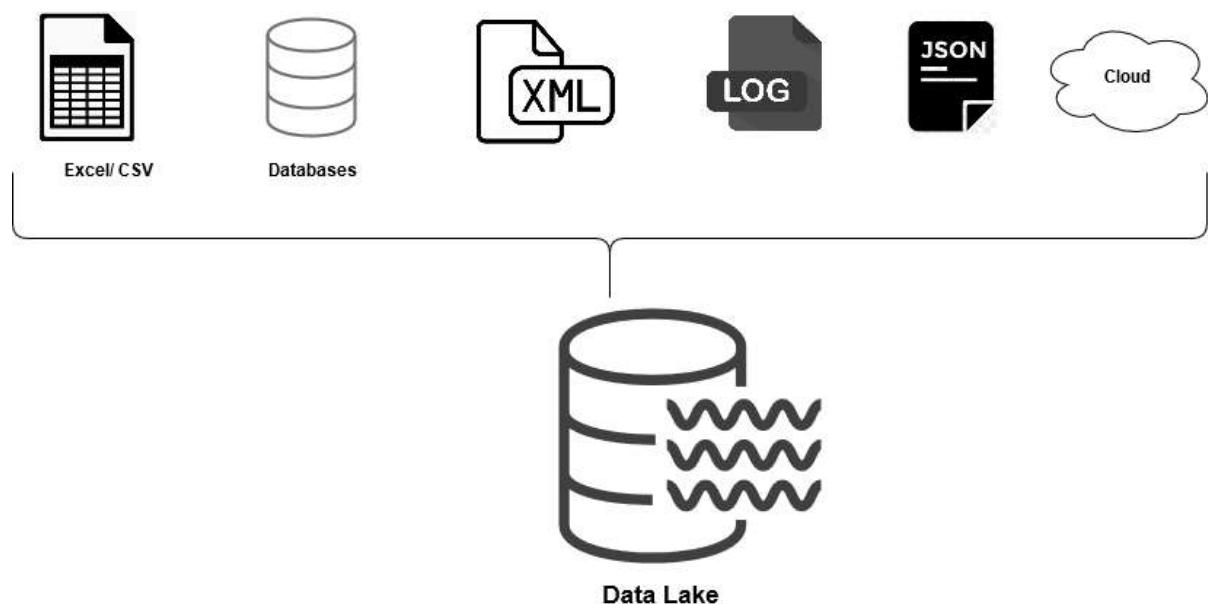
The data warehouse follows the Extract-Transform-Load mechanism for data transfer.

- Extract: Extract data from different data sources
- Transform: Transform data into a specific format
- Load: Load data into predefined data warehouse schema, tables



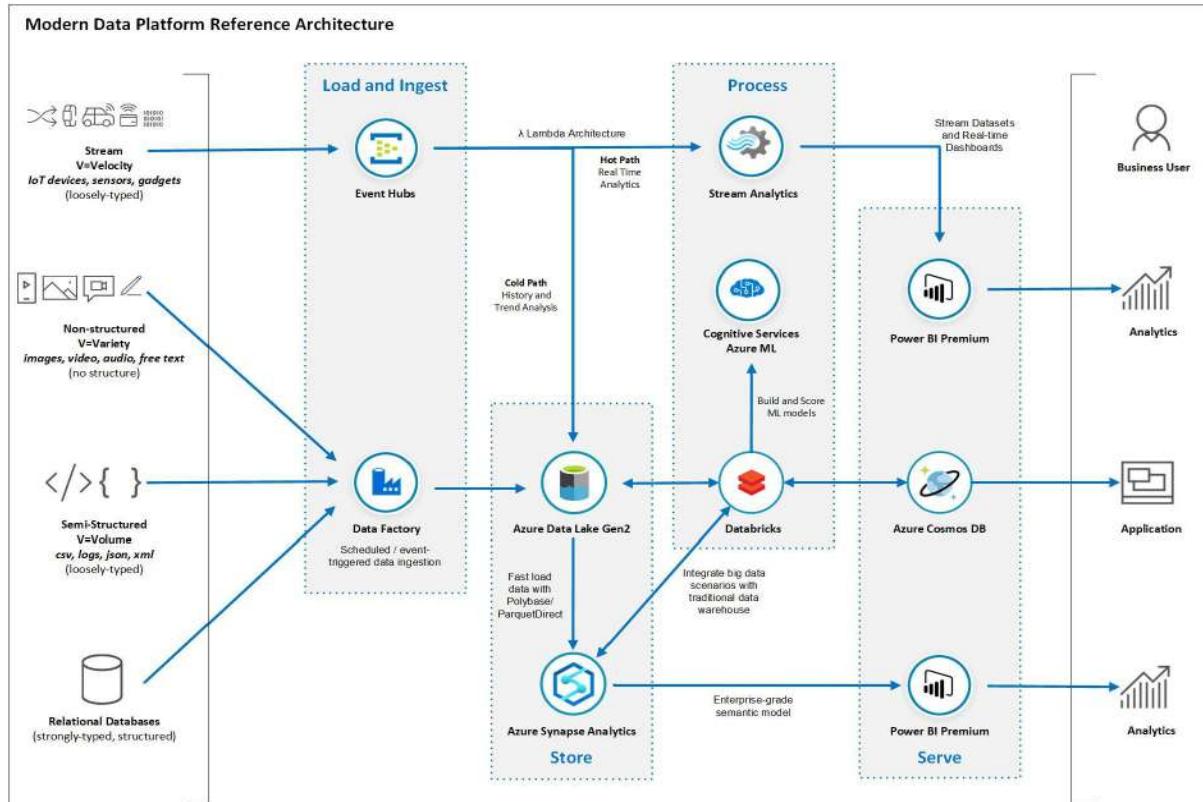
Various Data Sources

The data lake does not require a rigorous schema and converts data into a single format before analysis. It stores data in its original format such as binary, video, image, text, document, PDF, JSON. It transforms data only when needed. The data can be in structured, semi-structured and unstructured format.



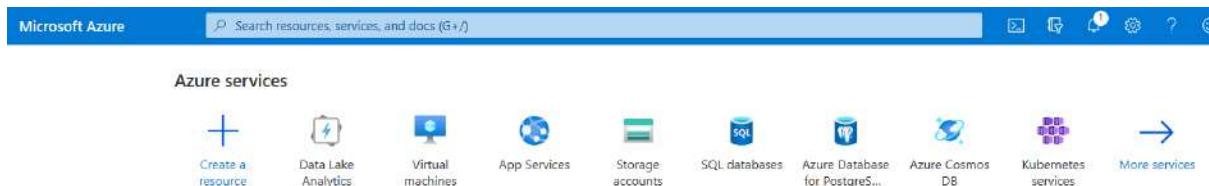
A few useful features of a data lake are:

- It stores raw data ( original data format)
- It does not have any predefined schema
- You can store Unstructured, semi-structured and structured in it
- It can handle PBs or even hundreds of PBs data volumes
- Data lake follows schema on the reading method in which data is transformed as per requirement basis
- **Ingestion:** Data collection from various data sources and store into the Azure Data lake in its original format
- **Storage:** Store data into Azure Data Lake Storage, AWS S3 or Google cloud storage
- **Processing:** Process data from the raw storage into a compatible format
- **Analytics:** Perform data analysis using stored and processed data. You can use Azure Data Lake Analytics(ADLA), HDInsight or Azure Databricks



# Creating an Azure Data Lake Analytics (ADLA) Account

We need to create an ADLA account with your subscription to process data with it. Login to the Azure portal using your credentials. In the Azure Services, click on Data Lake Analytics.



In the New Data Lake Analytics account, enter the following information.

- **Subscription and Resource group:** Select your Azure subscription and resource group, if it already exists. You can create a new resource group from the data lake analytics page as well
- **Data Lake Analytics Name:** Specify a suitable name for the analytic service
- **Location:** Select the Azure region from the drop-down
- **Storage subscription:** Select the storage subscription from the drop-down list
- **Azure Data Lake Storage Gen1:** Create a new Azure Data Lake Storage Gen1 account
- **Pricing package:** You can select a pay-as-you model or monthly commitment as per your requirement

## New Data Lake Analytics account

The Azure Data Lake Analytics service was architected from the ground up for cloud scale and performance. It takes away the complexities normally associated with big data in the cloud and ensures that Data Lake Analytics will meet your current and future business needs. [Learn more](#)

### Project details

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription \*

P3-Real Hands-On Labs

Resource group \*

1-116b5b07-playground-sandbox

[Create new](#)

### Data Lake Analytics details

Name

sqlshackdemo

sqlshackdemo.azuredatalakeanalytics.net

Location \*

East US 2

Existing storage subscription \*

P3-Real Hands-On Labs

Azure Data Lake Storage Gen1 \* ⓘ

(New) sqlshackdemostorage

[Create new](#)

Pricing package ⓘ

Pay-as-You-Go  Monthly commitment

[Review + create](#)

[Next](#)

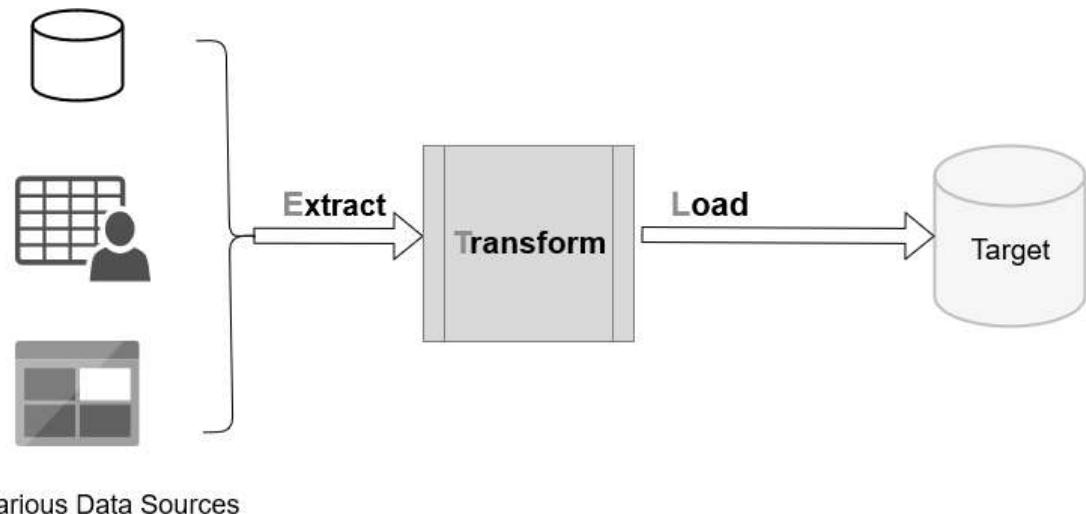
## Introduction to Azure Data Lake Analytics (ADLA)

Microsoft Azure platform supports big data such as Hadoop, HDInsight, Data lakes.

Usually, a traditional data warehouse stores data from various data sources, transform data into a single format and analyze for decision making. Developers use complex queries that might take longer hours for data retrieval. Organizations are increasing their footprints in the Cloud infrastructure. It leverages cloud infrastructure warehouse solutions such as Amazon RedShift, Azure Synapse Analytics (Azure SQL data warehouse), or AWS snowflake. The cloud solutions are highly scalable and reliable to support your data and query processing and storage requirements.

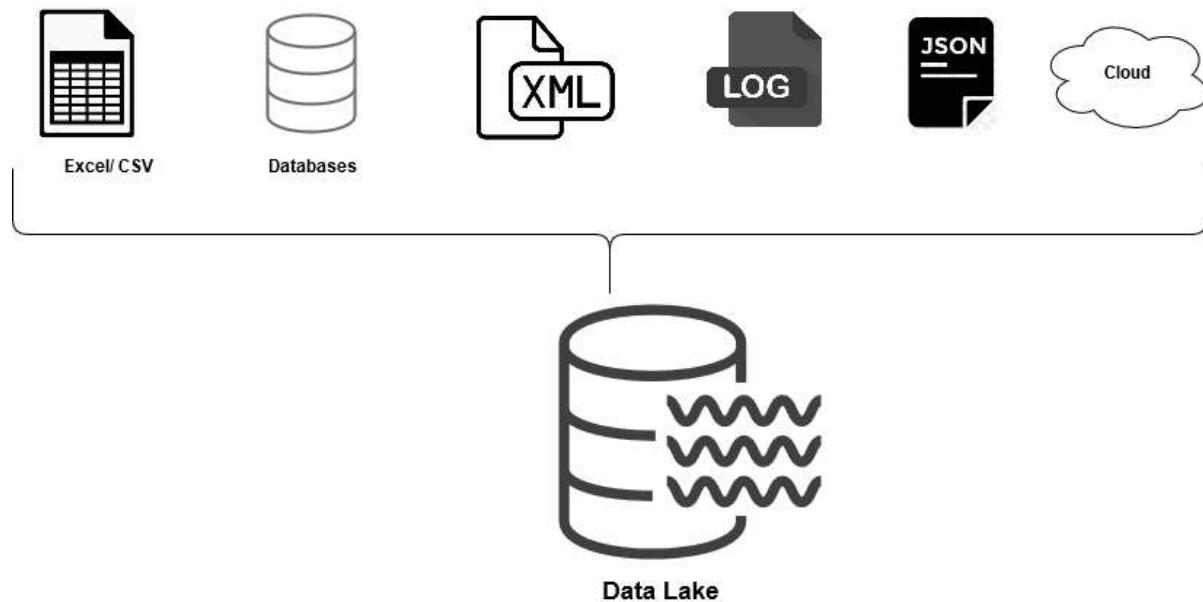
The data warehouse follows the Extract-Transform-Load mechanism for data transfer.

- **Extract:** Extract data from different data sources
- **Transform:** Transform data into a specific format
- **Load:** Load data into predefined data warehouse schema, tables



Various Data Sources

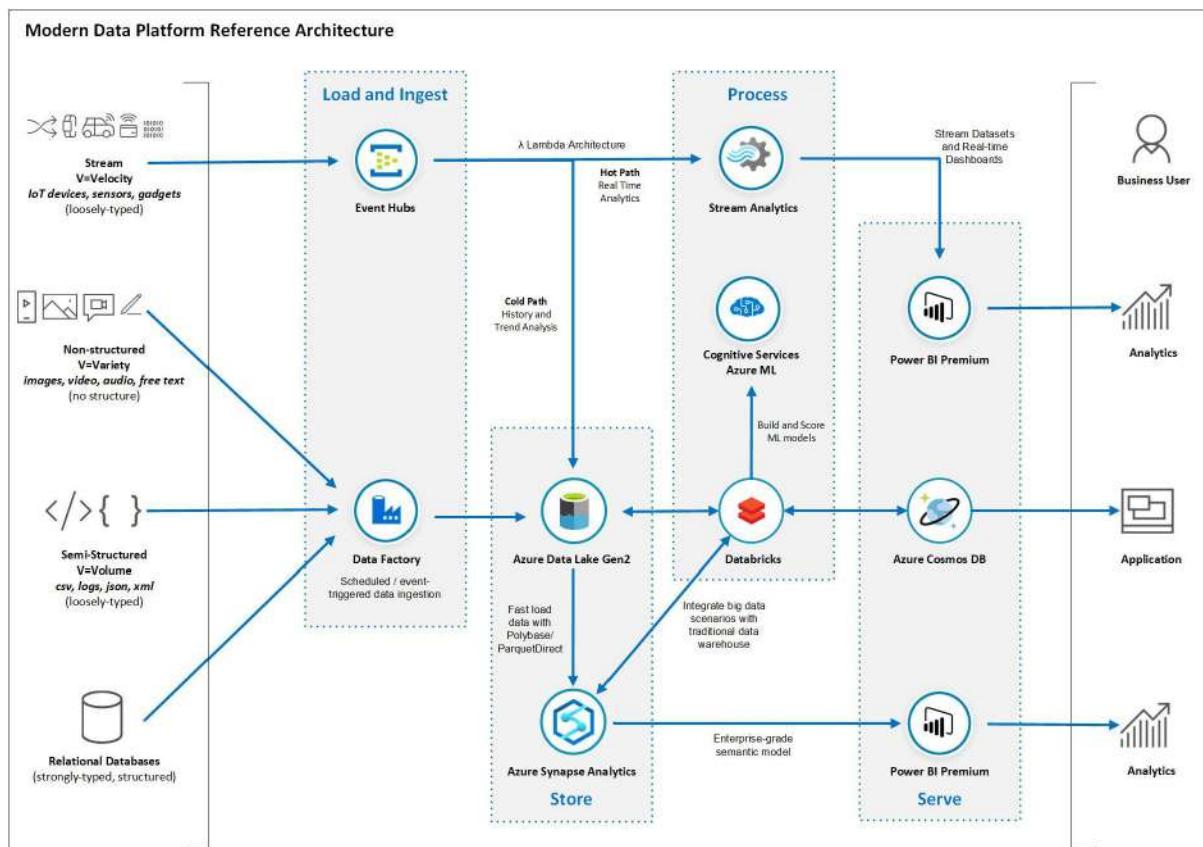
The data lake does not require a rigorous schema and converts data into a single format before analysis. It stores data in its original format such as binary, video, image, text, document, PDF, JSON. It transforms data only when needed. The data can be in structured, semi-structured and unstructured format.



A few useful features of a data lake are:

- It stores raw data ( original data format)
- It does not have any predefined schema
- You can store Unstructured, semi-structured and structured in it

- It can handle PBs or even hundreds of PBs data volumes
- Data lake follows schema on the reading method in which data is transformed as per requirement basis
- **Ingestion:** Data collection from various data sources and store into the Azure Data lake in its original format
- **Storage:** Store data into Azure Data Lake Storage, AWS S3 or Google cloud storage
- **Processing:** Process data from the raw storage into a compatible format
- **Analytics:** Perform data analysis using stored and processed data. You can use Azure Data Lake Analytics(ADLA), HDInsight or Azure Databricks



## Creating an Azure Data Lake Analytics (ADLA) Account

We need to create an ADLA account with your subscription to process data with it. Login to the Azure portal using your credentials. In the Azure Services, click on Data Lake Analytics.

In the New Data Lake Analytics account, enter the following information.

- **Subscription and Resource group:** Select your Azure subscription and resource group, if it already exists. You can create a new resource group from the data lake analytics page as well
- **Data Lake Analytics Name:** Specify a suitable name for the analytic service
- **Location:** Select the Azure region from the drop-down
- **Storage subscription:** Select the storage subscription from the drop-down list
- **Azure Data Lake Storage Gen1:** Create a new Azure Data Lake Storage Gen1 account
- **Pricing package:** You can select a pay-as-you model or monthly commitment as per your requirement

[Home](#) > [Data Lake Analytics](#) >

## New Data Lake Analytics account

The Azure Data Lake Analytics service was architected from the ground up for cloud scale and performance. It takes away the complexities normally associated with big data in the cloud and ensures that Data Lake Analytics will meet your current and future business needs. [Learn more](#)

### Project details

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription \*

P3-Real Hands-On Labs

Resource group \*

1-116b5b07-playground-sandbox

[Create new](#)

### Data Lake Analytics details

Name

sqlshackdemo

sqlshackdemo.azuredatalakeanalytics.net

Location \*

East US 2

Existing storage subscription \*

P3-Real Hands-On Labs

Azure Data Lake Storage Gen1 \* ⓘ

(New) sqlshackdemostorage

[Create new](#)

Pricing package ⓘ

Pay-as-You-Go  Monthly commitment

[Review + create](#)

[Next](#)

Click on Review+Create. You can review your configurations and create the Azure Data Lake Analytics Account.

[Home](#) > [Data Lake Analytics](#) >

## New Data Lake Analytics account

\* Basics    [Review + create](#)

[Summary](#)

New resources



Data Lake Analytics  
sqlshackdemo

Pricing tier  
Pay-as-You-Go



Data Lake Storage Gen1  
sqlshackdemostorage

Pricing tier  
Pay-as-You-Go

Basics

Subscription	P3-Real Hands-On Labs
Resource group	1-116b5b07-playground-sandbox
Location	East US 2
Pricing tier	Pay-as-You-Go
Storage account name	sqlshackdemostorage
Storage account type	Azure Data Lake Storage Gen1

---

[Create](#)

[Previous](#)

## Data Lake Analytics pricing

- No upfront cost
- No termination fees
- Pay only for what you use
- Per-second billing

## Start in seconds, scale instantly, pay per job



Process big data jobs in seconds with Azure Data Lake Analytics. There is no infrastructure to worry about because there are no servers, virtual machines or clusters to wait for, manage or tune.

Instantly scale the processing power, measured in Azure Data Lake Analytics Units (AU), from one to thousands for each job. You only pay for the processing which you use per job.

## Develop massively parallel programs with simplicity



U-SQL is a simple, expressive and extensible language which allows you to write code once and have it automatically parallelised for the scale you need. Process petabytes of data for diverse workload categories such as querying, ETL, analytics, machine learning, machine translation, image processing and sentiment analysis by leveraging existing libraries written in .NET languages, R or Python.

## **Virtualise your analytics**



Act on all of your data with optimised data virtualisation of your relational sources such as Azure SQL Database and Azure Synapse Analytics. Your queries are automatically optimised by moving processing close to the source data without data movement, which maximises performance and minimises latency.

## **Enterprise-grade security, auditing and support**



Extend your on-premises security and governance controls to the cloud and meet your security and regulatory compliance needs. Single sign-on (SSO), multi-factor authentication and seamless management of millions of identities are built-in through Azure Active Directory. Role-based access control and the ability to audit all processing and management operations are on by default. We guarantee a 99.9% enterprise-grade SLA and 24/7 support for your big data solution.

# AZURE DATA BRICKS SERVICE



## WHAT IS DATA BRICKS?

Databricks was originally developed by the creators of Apache Spark and aims to deliver a unified platform where data scientists & engineers can work together to build end-to-end machine learning solutions from data discovery up to production.

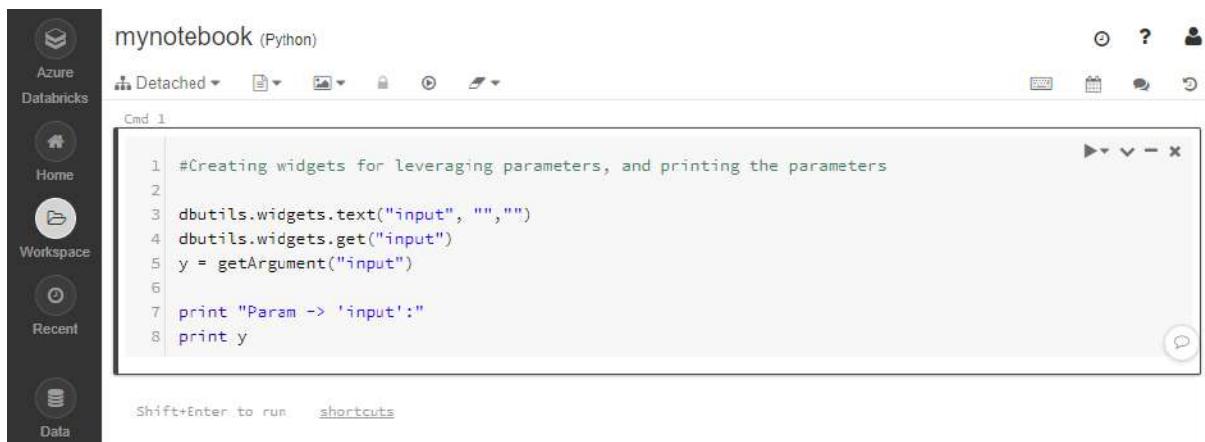
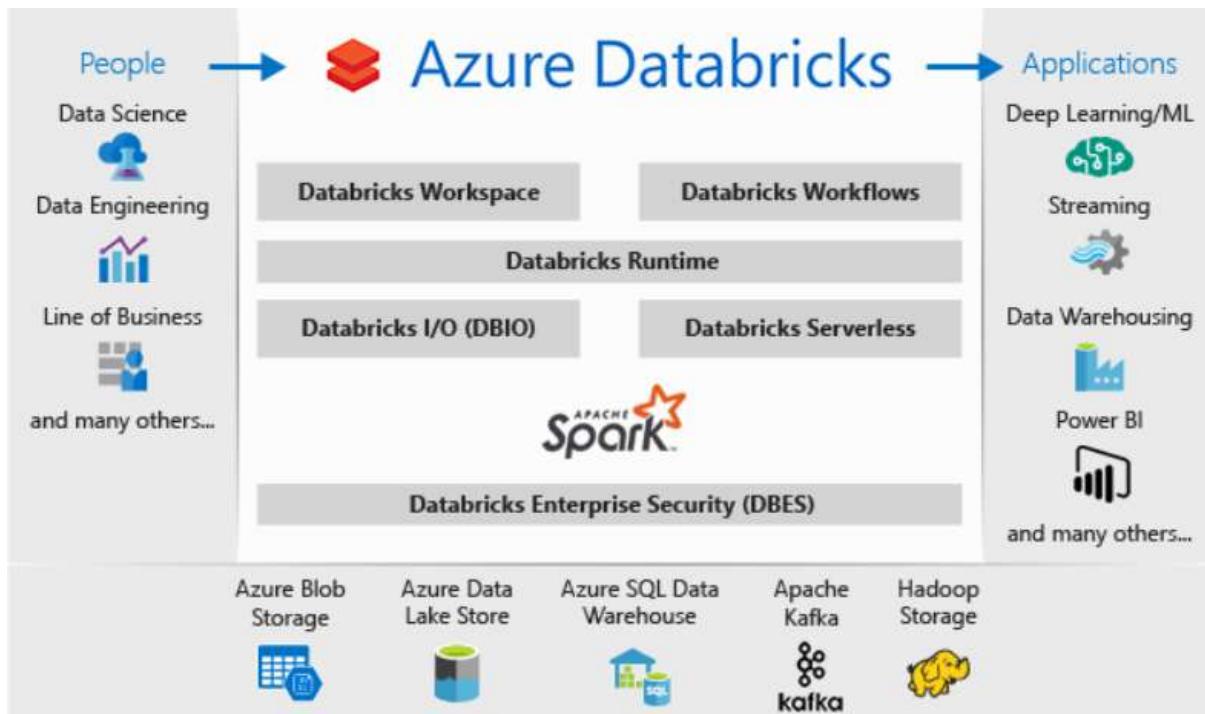
Databricks is a platform where users can log in & work. It's built on top of Apache Spark computing technology & can be mounted on-premise or in a Cloud set-up giving the users any needed compute power to work in an abstracted and simplified way.

Azure Databricks offers all the components and capabilities of Databricks Apache Spark with a possibility to integrate it with other Microsoft Azure services.

## WHAT IS AZURE DATA BRICKS SERVICE?

Designed together with Microsoft, Azure Databricks is a managed version of Databricks that gives Azure customers the ability to do one-click set up, streamlined workflows and shared collaborative interactive workspaces.

It enables fast collaboration between data scientists, data engineers, and business analysts through the Databricks platform. Azure Databricks is tightly connected with Azure storage & compute resources such as Azure Blob Storage, Data Lake Store, SQL Data Warehouse & HDInsights.



## Azure Databricks Workspace

Azure Databricks has a support for Python, Scala, R and SQL and some libraries for deep learning like Tensorflow, Pytorch and Scikit-learn for building big data analytics and AI solutions. In Azure Databricks notebooks, the user can easily switch between different

programming languages with just simple language commands to make use of more languages in one notebook.

Running a job on the cluster in Azure Databricks, means running a notebook, either manually or by scheduling it to run at a specific time. Azure Databricks provides different users in the organization the possibility to collaborate on shared projects in one workspace.

The screenshot shows the Azure Databricks landing page. At the top left is the Databricks logo and the text "Azure Databricks". The page is divided into three main sections:

- Explore the Quickstart Tutorial**: Shows a document icon with a lock symbol. Text: "Spin up a cluster, run queries on preloaded data, and display results in 5 minutes."
- Import & Explore Data**: Shows a dashed box with a "Drop files or click to browse" placeholder and a cloud icon. Text: "Quickly import data, preview its schema, create a table, and query it in a notebook."
- Create a Blank Notebook**: Shows a document icon with a plus sign. Text: "Create a notebook to start querying, visualizing, and modeling your data."

Below these sections are three cards:

- Common Tasks**: Includes icons for New Notebook, Upload Data, Create Table, New Cluster, New Job, Import Library, and Read Documentation.
- Recents**: Text: "Recent files appear here as you work."
- Documentation**: Includes links to Databricks Guide, Python, R, Scala, SQL, and Importing Data.

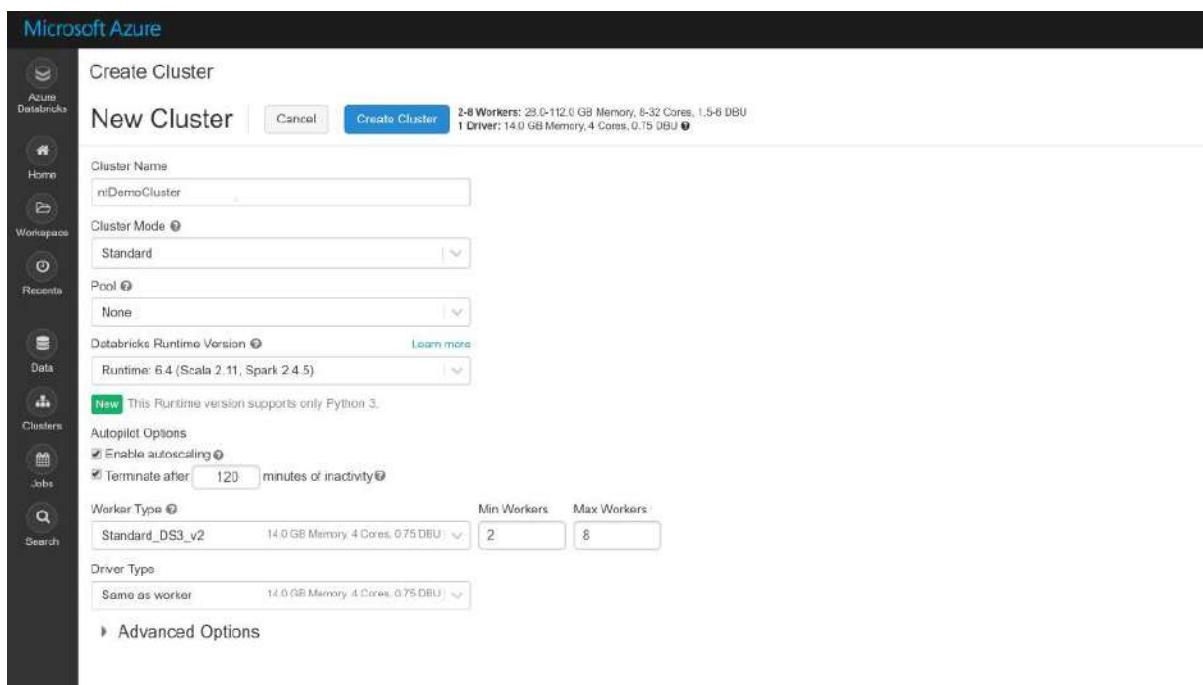
# Big data analytics and AI with optimised Apache Spark

Unlock insights from all your data and build artificial intelligence (AI) solutions with Azure Databricks, set up your Apache Spark™ environment in minutes, autoscale and collaborate on shared projects in an interactive workspace. Azure Databricks supports Python, Scala, R, Java and SQL, as well as data science frameworks and libraries including TensorFlow, PyTorch and scikit-learn.



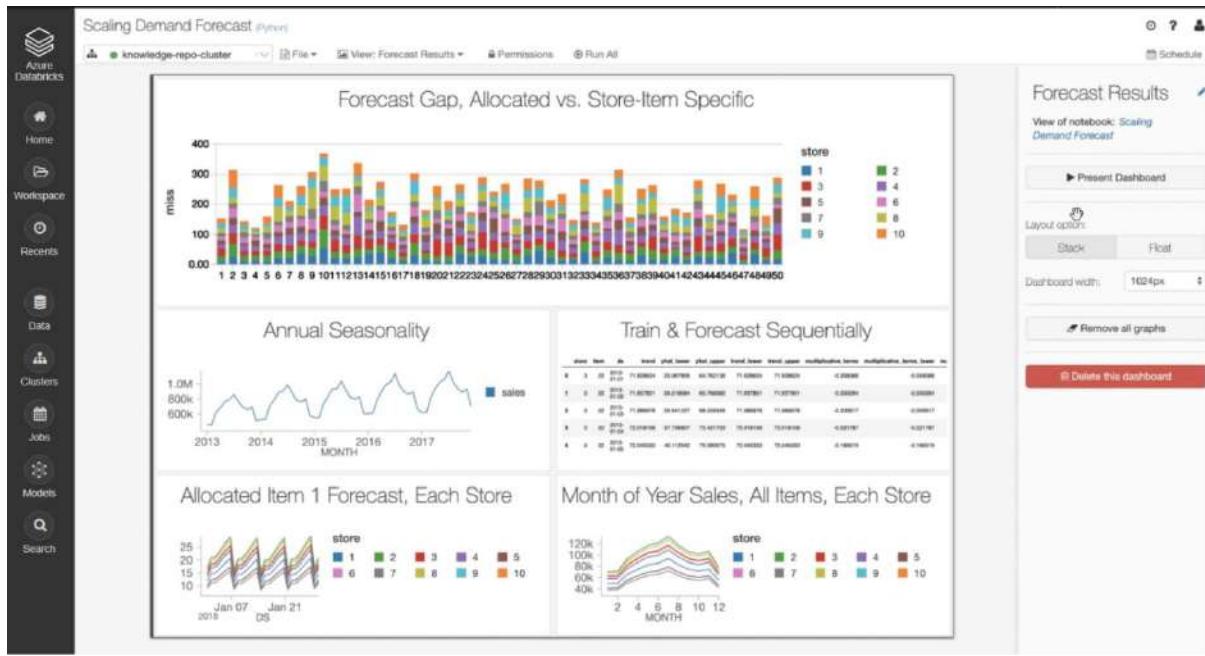
## Start quickly with an optimised Apache Spark environment

Azure Databricks provides the latest versions of Apache Spark and allows you to seamlessly integrate with open source libraries. Spin up clusters and build quickly in a fully managed Apache Spark environment with the global scale and availability of Azure. Clusters are set up, configured and fine-tuned to ensure reliability and performance without the need for monitoring. Take advantage of autoscaling and auto-termination to improve total cost of ownership (TCO).



## Get high-performance modern data warehousing

Combine data at any scale and get insights through analytical dashboards and operational reports. Automate data movement using Azure Data Factory, then load data into Azure Data Lake Storage, transform and clean it using Azure Databricks and make it available for analytics using Azure Synapse Analytics. Modernise your data warehouse in the cloud for unmatched levels of performance and scalability.



## KEY SERVICE CAPABILITIES:

### Optimised spark engine

Simple data processing on autoscaling infrastructure, powered by highly optimised Apache Spark™ for up to 50 x performance gains.

### Machine learning run time

One-click access to preconfigured machine learning environments for augmented machine learning with state-of-the-art and popular frameworks such as PyTorch, TensorFlow and scikit-learn.

### MLflow

Track and share experiments, reproduce runs and manage models collaboratively from a central repository.

### Choice of language

Use your preferred language, including Python, Scala, R, Spark SQL and .Net—whether you use serverless or provisioned compute resources.

## **Collaborative notebooks**

Quickly access and explore data, find and share new insights and build models collaboratively with the languages and tools of your choice.

## **Delta lake**

Bring data reliability and scalability to your existing data lake with an open source transactional storage layer designed for the full data lifecycle.

## **Native integrations with Azure services**

Complete your end-to-end analytics and machine learning solution with deep integration with Azure services such as Azure Data Factory, Azure Data Lake Storage, Azure Machine Learning and Power BI.

## **Interactive workspaces**

Enable seamless collaboration between data scientists, data engineers and business analysts.

## **Enterprise-grade security**

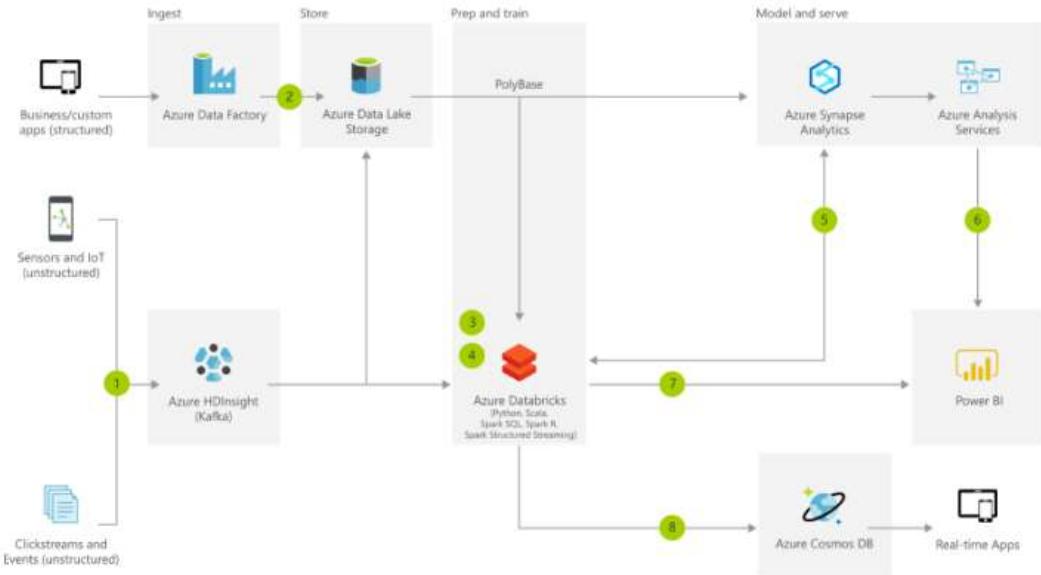
Effortless native security protects your data where it lives and creates compliant, private and isolated analytics workspaces across thousands of users and datasets.

## **Production-ready**

Run and scale your most mission-critical data workloads with confidence on a trusted data platform, with ecosystem integrations for CI/CD and monitoring.

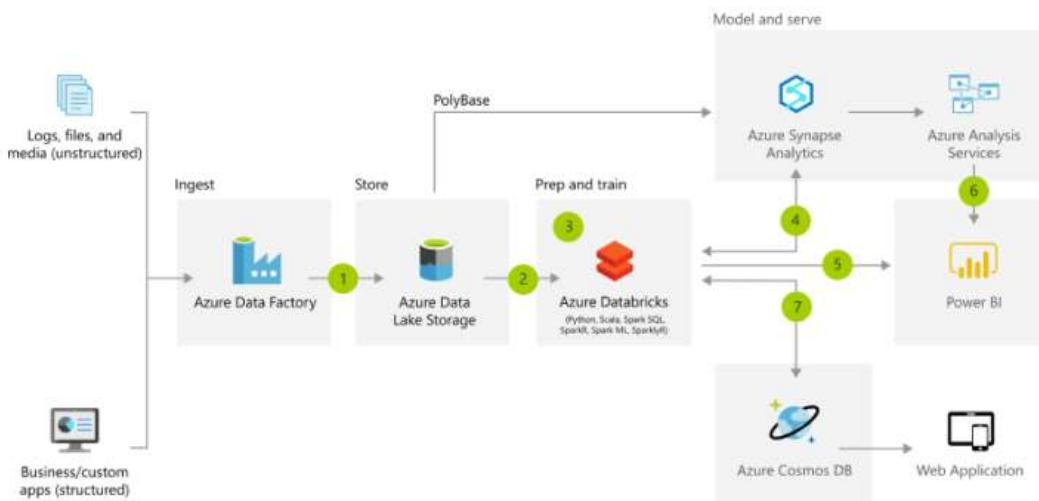
## **Data science and machine learning with Azure Databricks**

Get insights from live-streaming data with ease. Capture data continuously from any IoT device or logs from website clickstreams and process it in near-real time.



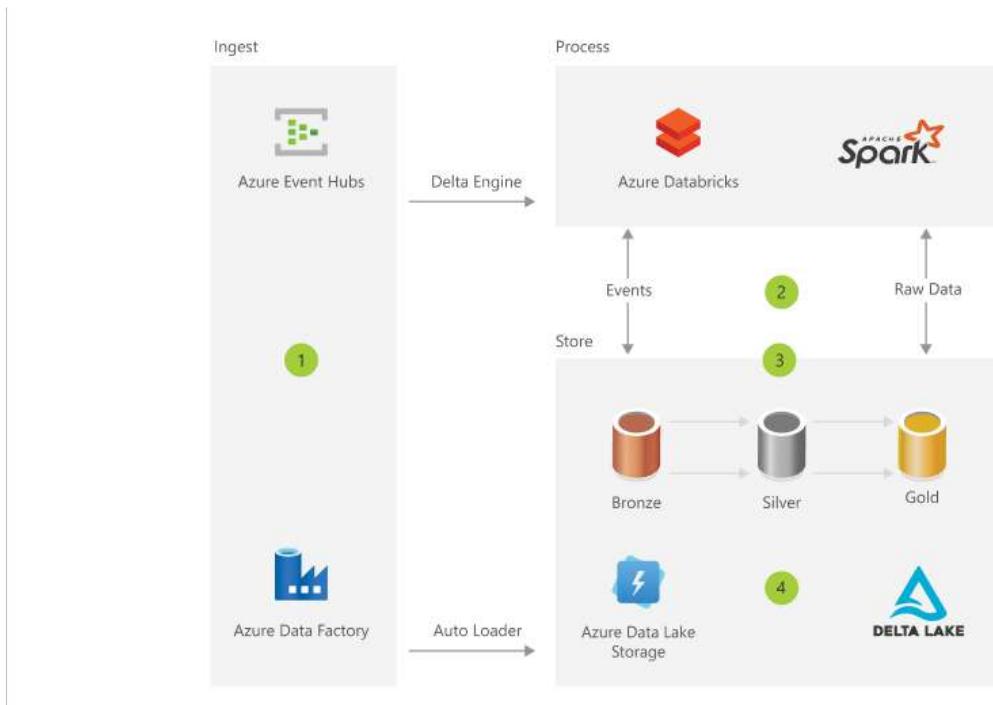
## Modern analytics architecture with Azure Databricks

Transform your data into actionable insights using best-in-class machine learning tools. This architecture allows you to combine any data at any scale and to build and deploy custom machine learning models at scale.



## Ingest, ETL and stream processing pipelines with Azure Databricks

Accelerate and manage your end-to-end machine learning lifecycle with Azure Databricks, MLflow and Azure Machine Learning to build, share, deploy and manage machine learning applications.

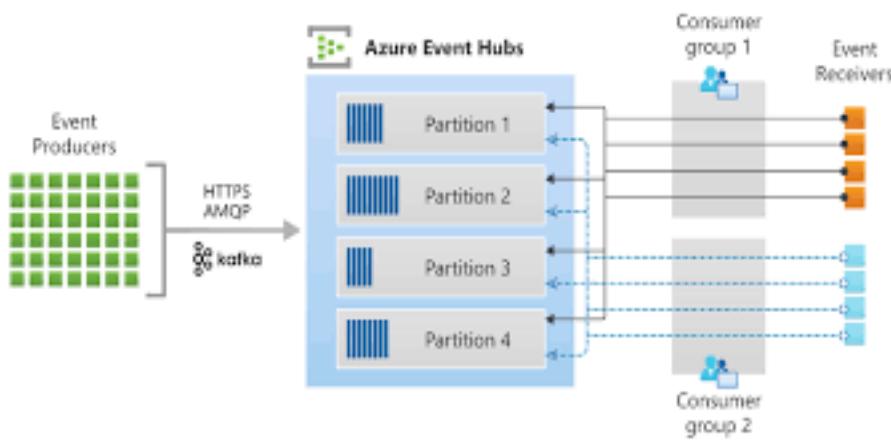


# AZURE EVENT HUBS SERVICE



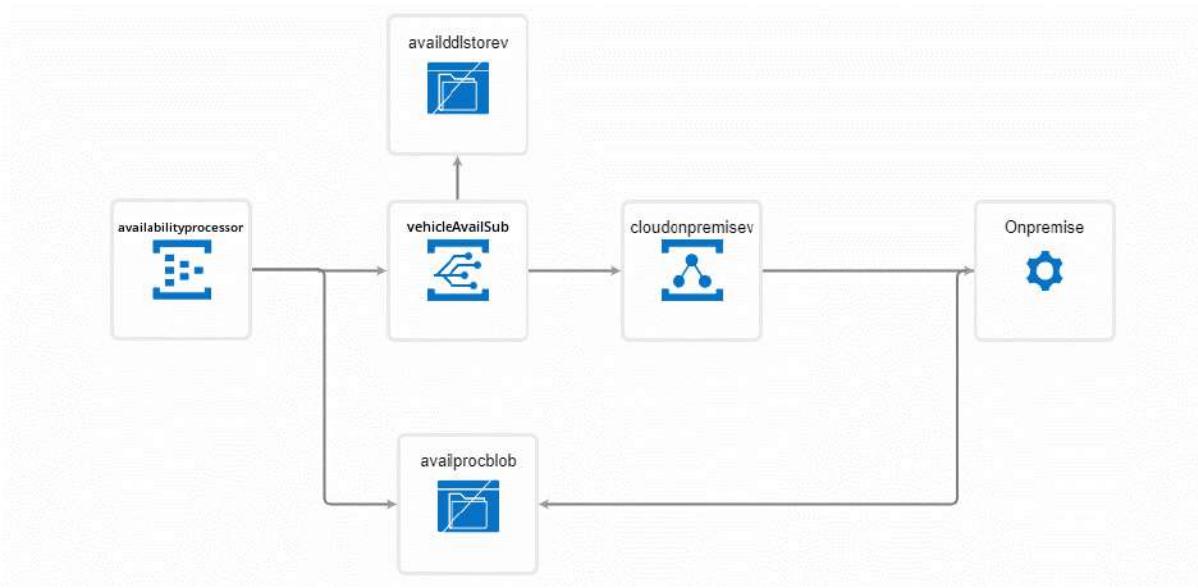
## WHAT IS AZURE EVENT HUB?

Azure Event Hubs is a big data streaming platform and event ingestion service. It can receive and process millions of events per second. Data sent to an event hub can be transformed and stored by using any real-time analytics provider or batching/storage adapters.



## What is Azure Event Hub used for?

Azure Event hubs and Event Grid are used in business orchestrations for event storage and handling. Let's take a simple Vehicle availability management scenario for better understanding.

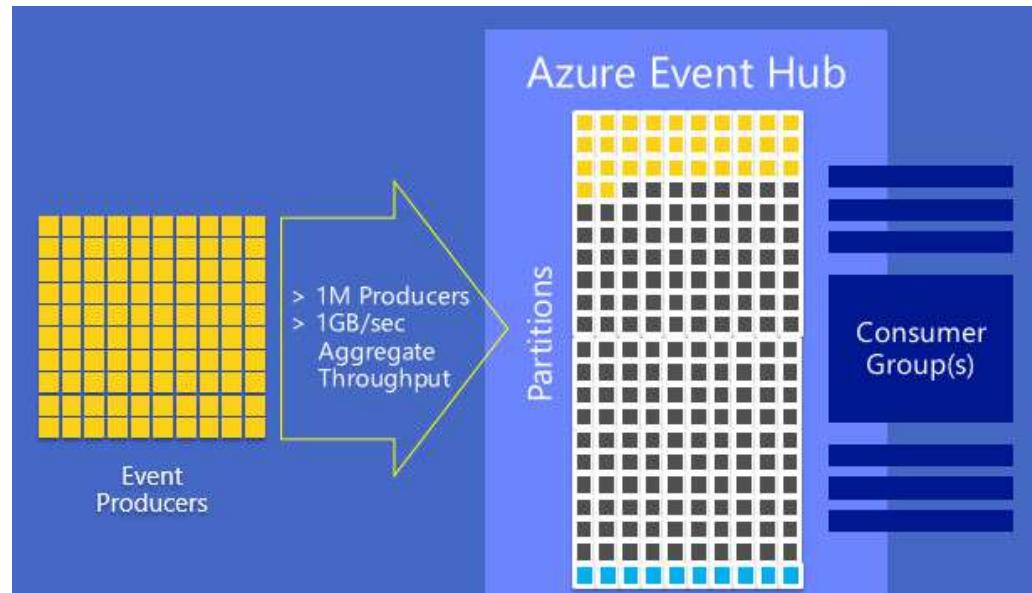


In the orchestration above, the event hub capture feature automatically writes batches of captured events into the Azure Storage Blob containers and enables timely batch-oriented processing of events.

Azure Event Hubs emits an event to the Event Grid when the capture file is created. These events are not strongly correlated and don't require processing in batches. Hence, Event Grid is selected to provide a reliable event delivery at massive scale.

Event Grid delivers the event to the Azure Relay which securely exposes the service that runs in the corporate network to the public cloud. Therefore, the actual business logic to process the telematics data in the storage blob container for decision-making analytics resides in the on-premise service.

## What is Event Hub Partitions?



Azure Event Hub splits up the streaming data into partitions in which the consumer or receiver will be restricted only to a specific subset or partition of the data. The storage and retrieval of the data are different using the partitions, unlike Service Bus queues and topics. In Service Bus queues and topics each consumer reads from the single queue lane. It is modelled based on “Competing Consumer” pattern where the single queue lane method will end up in scale limits. In contrast to the Service bus modelling, Azure Event Hubs is based on the “Partitioned Consumer” pattern where multiple lanes are assigned which boosts up the scaling.

## Azure Event Grid Vs Event Hub

As discussed before, an event is a lightweight communication method. To make the best use of it Azure brings in two event-based technologies. Let's see the contrasted capabilities showcased by the Event Grid and Event Hub with real-time scenarios that fit in.

### Azure Event Hub



Azure Event hubs – Event Ingesting service

Azure Event Hub is a data ingestion service which streams a huge count of messages from any source to provide an immediate response to business challenges. The event Hub comes into play when handling of events along with data is required. Unlike Event Grids, Event Hubs perform certain additional tasks apart from just being an event broadcaster.

### Azure Event Grid



Event Grid Topic – Event Publisher



#### Event Grid Subscription – Event Subscriber

Event Grid is an event-based technology which allows the publisher to inform the consumer regarding any status change. It is an event routing service. This service acts as a connector to tightly bind all the applications together and routes the event messages from any source to any destination.

## Geo-Disaster Recovery

The downfall is a common pitch faced by any cloud provider. Similarly, Azure regions or datacentres encounters a downfall if no availability zones are used. Due to the fall in datacentres, data processing gets affected and switching the processing into a completely different region or datacentre seems critical. To overcome the disrupt Geo-disaster recovery and Geo-replication comes into action, which are important features for any enterprise. Azure Event hubs support both geo-disaster recovery and geo-replication at the same namespace level.

The Geo-disaster recovery is feasible only for standard and dedicated SKUs. The Geo-disaster recovery is feasible only for standard and dedicated SKUs.

## Security

Security is one of the important parameters in any resource. Azure Event hubs provide security at two levels,

- Authorization
- Authentication

## Authorization

Every request to a secure resource must be authorized to ensure that the client has required permissions to access the data. Azure Event Hubs provide few options for authorizing access through,

- [Azure Active Directory](#)
- Shared Access Signature

## Authentication

Authentication is the process of validating and verifying the identity of a user or process. The user authentication ensures that the individual is recognised by the Azure platform. User verification can be done by,

- Authenticating with Azure Active Directory
- Authenticating with Shared Access Signature

## Security Controls

Security control is a characteristic of an Azure resource that inculcates the ability to prevent, observe and respond to security defects. Azure Event Hubs possess few security controls in various perspectives,

Network	Service endpoint support Network isolation and fire walling support
Monitoring and Logging	Azure monitoring support Control and management plane logging and audit Data plane logging and audit
Identity	Authentication Authorization
Data Protection	Microsoft-managed keys Customer-managed keys Encryption in transit API calls encryption

## How do I read data from Azure Event Hub?

Data from the Azure Event hub can be received using an Event hub consumer. Multiple consumers can be allocated for the same event hub, they can read the same partition data at their tempo. Event hub consumers are connected through AMQP channels, this makes data availability easier for clients.

## How do I connect to Event Hub?

Shared Access Policy can be used to connect an application with Azure Event hub. To obtain the Shared Access Keys, we can use the Azure portal or Azure CLI.

## How do I create an Event Hub in Azure?

- Sign-in to the Azure Portal.
- On the portal, click +New->Internet of Things->Event Hubs.
- In the “Create Namespace” blade, enter the name of your Event Hub in the name field, then choose the Standard Pricing Tier, and choose the desired subscription to create the Event Hub under it.

## How to send data to Azure Event Hub?

Follow the steps to send data to event hub via code,

- Create a console application in .NET Core.

- After you've finished the programme, you'll need to establish a Class Library project to submit the data. The main project or the driver, on the other hand, merely invokes functions from that class library.
- Create a “Sender” class Library project.
- Make a new file called Sample.Sender.cs. Make a folder called Models. This folder will house the class model that we will use to map our data before sending it. Create a SampleData class in the Models folder.
- Now add a class to the main driver project named SenderHelper.cs.
- The Main method will utilise this class as a controller to deliver data to the Event Hub
- In the Program.cs code mention the number of messages you would like to send to the Event Hub.

## How do I turn off an Azure Event Hub?

- Log in to the administrator interface for Azure Stack Hub.
- On the left, click Marketplace Management.
- Providers of resources should be chosen.
- From the list of resource providers, choose Event Hubs. You can narrow down the results by typing “Event Hubs” into the search box given.
- Choose Uninstall from the drop-down menus at the top of the page.
- Select Uninstall after entering the resource provider’s name.

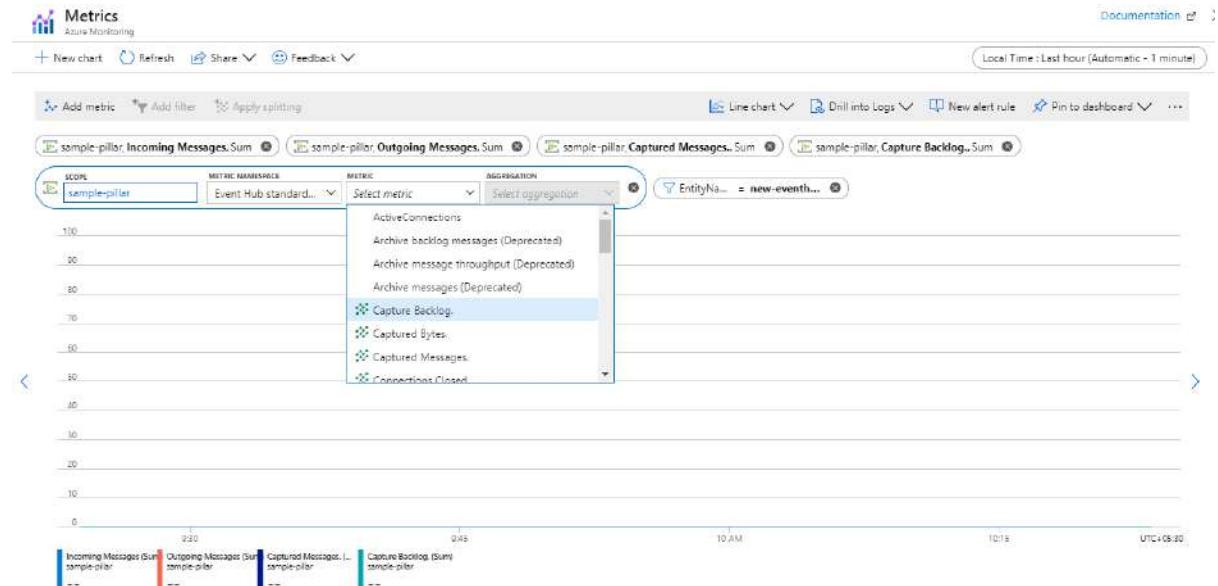
## How to access the metrics?

The metrics can be accessed either through the Azure portal or using APIs and Log Analytics. Metrics are enabled by default and most recent data of 30 days can be accessed or to retain data for a longer duration, the metrics data can be archived to an Azure Storage account.

Metrics can be monitored straight from the Azure portal directly from the namespace level just by clicking on the metrics option under monitoring section available in the left blade.

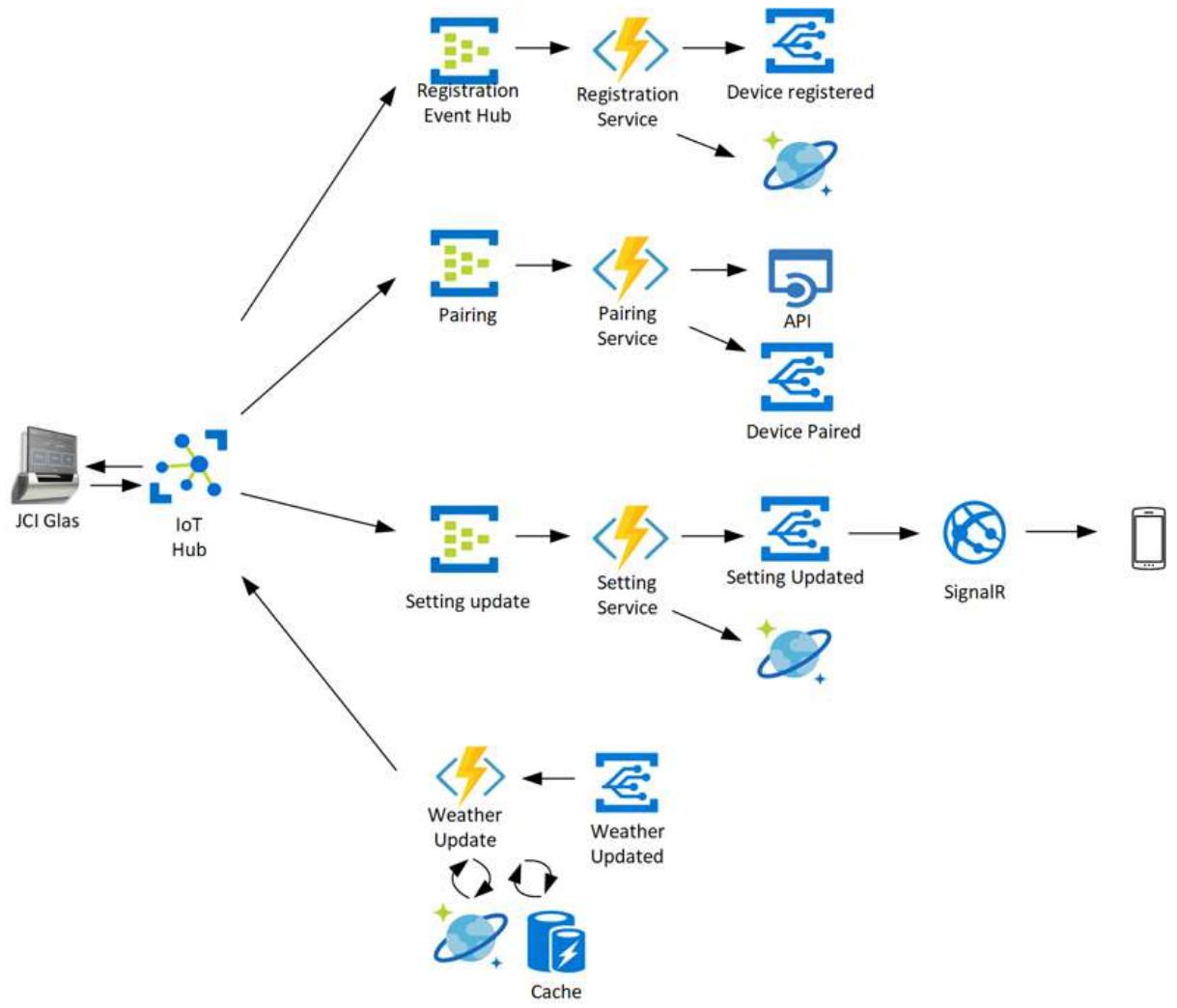


To bring in the required metrics for monitoring, specify the desired metric namespace and select from the metrics filtered to the scope of that event hub.



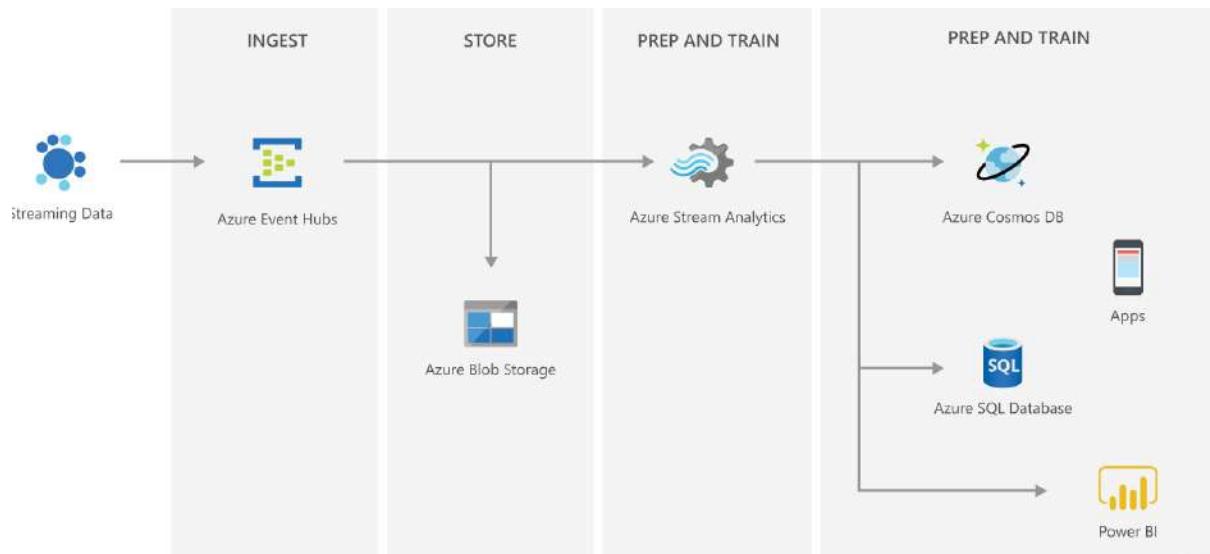
## Architecture

In this architecture, you can see four different activities, all controlled by different Azure Serverless components like Azure Event Hubs, Functions, Event Grid, Cosmos DB, and API Management. All the communication between the thermostat, Azure and the customer device is sent through the IoT Hub. As a first step when the customer installs a thermostat, JCI GLAS device sends a message to the IoT Hub to register the device. This event triggers the Functions app to save the data in Cosmos DB. The function app also raises an event into the Event Grid to trigger events like the registration success message to the customer mobile devices.



## Serverless streaming with Event Hubs

Build an end-to-end serverless streaming platform with Event Hubs and Stream Analytics



## Why choose Event Hubs?

Focus on drawing insights from your data instead of managing infrastructure. Build real-time big data pipelines and respond to business challenges right away.

- Simple

Build real-time data pipelines with just a couple clicks. Seamlessly integrate with Azure data services to uncover insights faster.

- Secure

Protect your real-time data. Event Hubs is certified by CSA STAR, ISO, SOC, GxP, HIPAA, HITRUST and PCI.

- Scalable

Adjust throughput dynamically based on your usage needs and pay only for what you use.

- Open

Ingest data from anywhere and develop across platforms with support for popular protocols, including AMQP, HTTPS and Apache Kafka®.

# AZURE HINDSIGHT SERVICE



## WHAT IS AZURE HDINSIGHT?

Azure HDInsight is a managed, full-spectrum, open-source analytics service in the cloud for enterprises. With HDInsight, you can use open-source frameworks such as Hadoop, Apache Spark, Apache Hive, LLAP, Apache Kafka, Apache Storm, R, and more, in your Azure environment.

## Why should I use Azure HDInsight?

- 1) Low Cost and Scalable
- 2) Secure and compliant
- 3) Monitoring
- 4) Global Activity
- 5) Productivity
- 6) Extensibility

## Scenarios for using HDInsight

Azure HDInsight can be used for a variety of scenarios in big data processing. It can be historical data (data that's already collected and stored) or real-time data (data that's

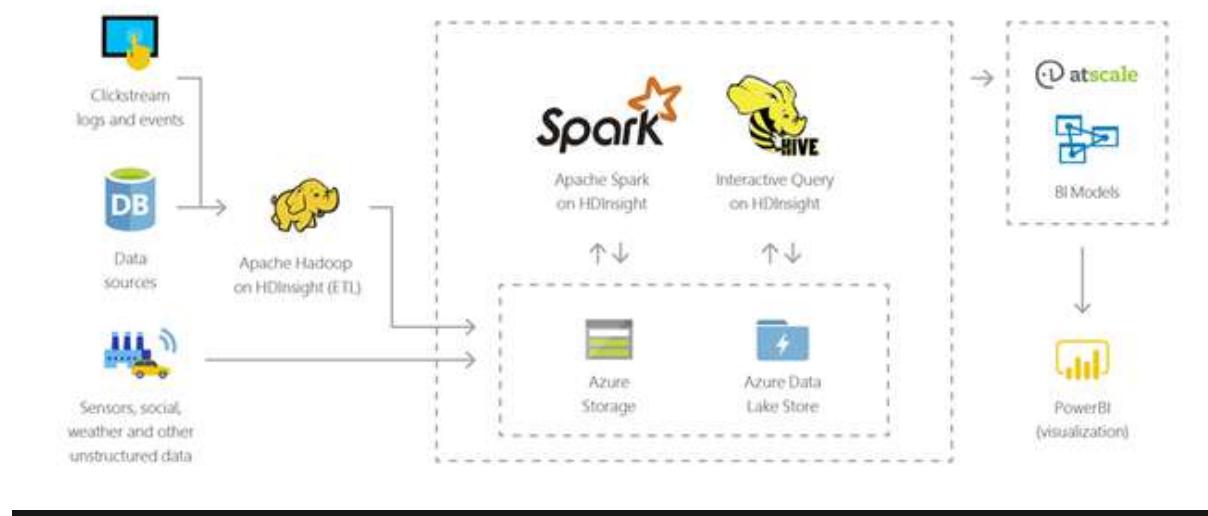
directly streamed from the source). The scenarios for processing such data can be summarized in the following categories:

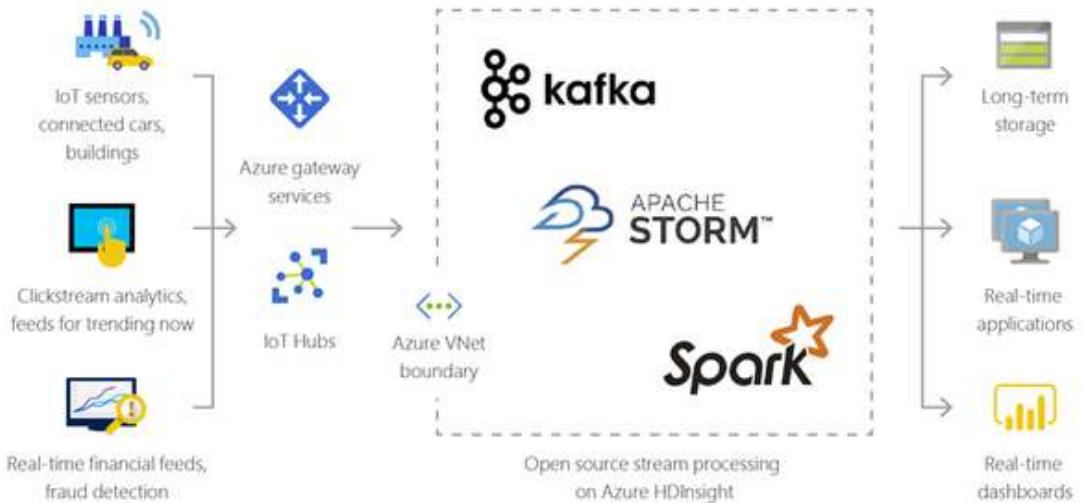
### Batch processing (ETL)

Extract, transform, and load (ETL) is a process where unstructured or structured data is extracted from heterogeneous data sources. It's then transformed into a structured format and loaded into a data store. You can use the transformed data for data science or data warehousing.

### Data warehousing

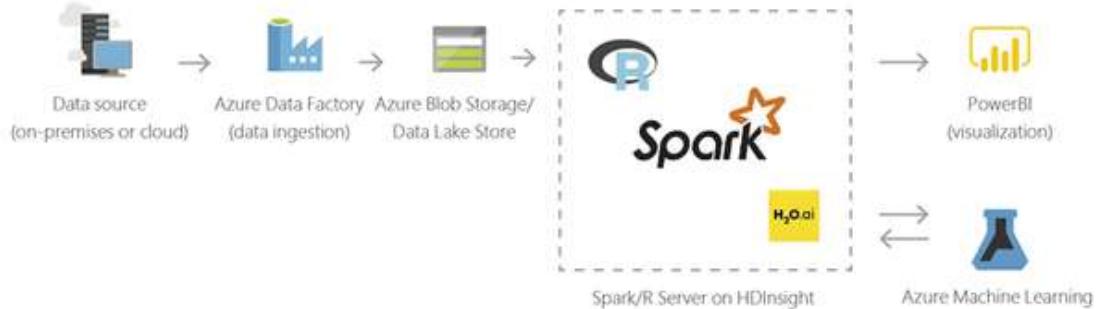
You can use HDInsight to perform interactive queries at petabyte scales over structured or unstructured data in any format. You can also build models connecting them to BI tools.





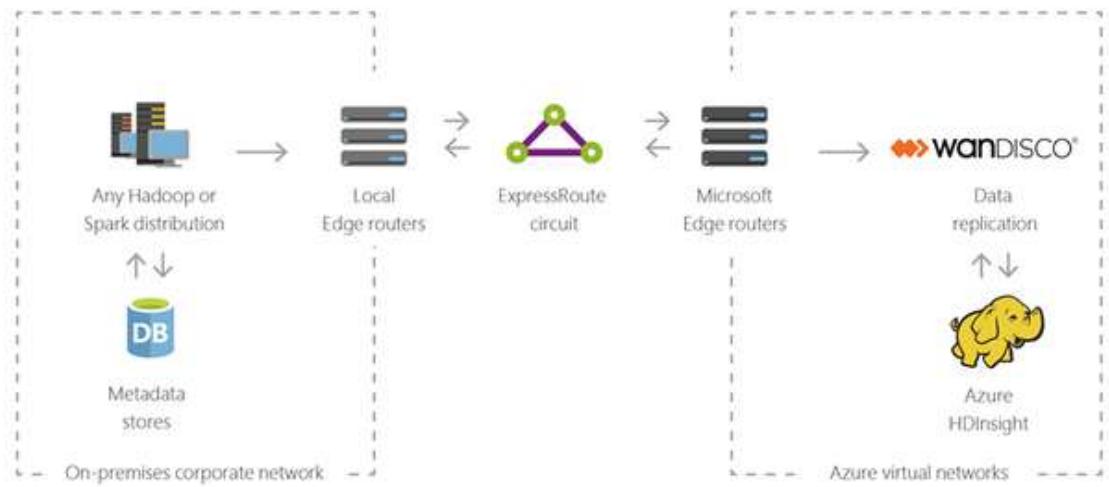
## Data science

You can use HDInsight to build applications that extract critical insights from data. You can also use Azure Machine Learning on top of that to predict future trends for your business. For more information, [read this customer story](#).



## Hybrid

You can use HDInsight to extend your existing on-premises [big data](#) infrastructure to Azure to leverage the advanced analytics capabilities of the cloud.



## Build your projects in an open-source ecosystem

Stay up to date with the newest releases of open source frameworks, including Kafka, HBase and Hive LLAP. HDInsight supports the latest open-source projects from the Apache Hadoop and Spark ecosystems.



## Azure HDInsight is trusted by companies of all sizes

### 1) Myntra

Mynta accelerates its digital transformation. Myntra has worked closely with Microsoft to migrate its platform—from supply chain management to inventory to site capabilities to Azure for trusted, always-on, hyperscale and cost-effective computing.



### 2) GAP

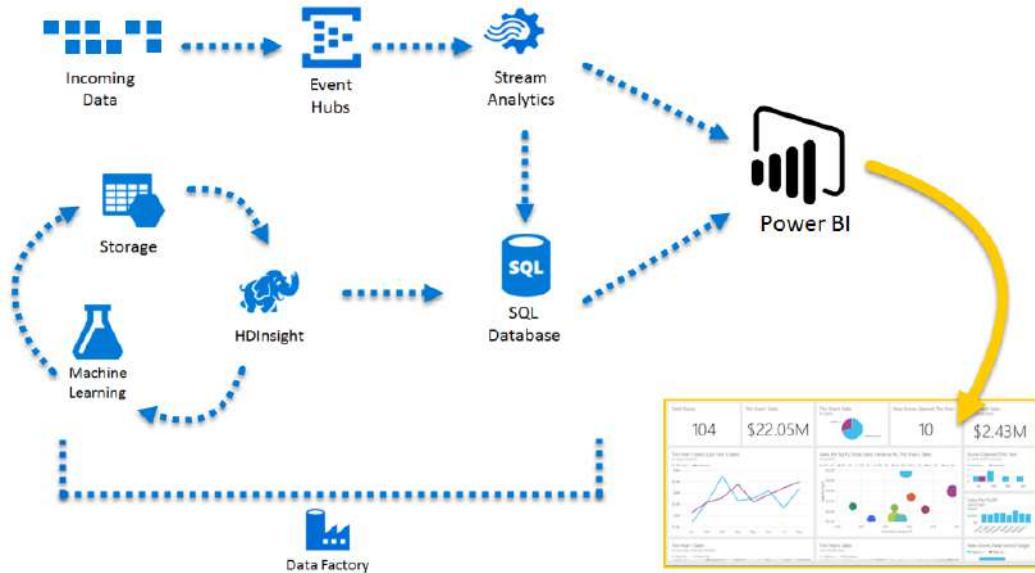
- 3) Whitehall Resources Limited
- 4) Lovren Technologies
- 5) Confidential Records, Inc

# AZURE POWER BI EMBEDDED SERVICE



## What is “POWER BI EMBEDDED” in Azure?

Quickly and easily provide customer-facing reports, dashboards and analytics in your own applications by using and branding as your own. Reduce developer resources by automating the monitoring, management, and deployment of analytics, while getting full control of Power BI features and intelligent analytics.



## USES:

Adopt decades of analytics expertise and access the continued investment Microsoft makes in analytics and AI



Choose the best way to visualise your data with out-of-the-box, certified and custom-built visuals



With visualisations optimised for desktop and mobile, your users can easily make decisions from anywhere



Pay as little as \$1/hour for analytics and scale as your business grows—with no requirement for end-user licensing



# Create Power BI Embedded capacity in the Azure portal

## Create a capacity

for creating a Power BI Embedded capacity, make sure you have signed into Power BI at least once.

- [Portal](#)
- [Azure CLI](#)
- [ARM template](#)
  1. Sign into the [Azure portal](#).
  2. Under **Azure services**, select *Power BI Embedded*.
  3. Within Power BI Embedded, select **Create**.
  4. Fill in the required information and then select **Review + Create**.

Power BI Embedded ...

\* Basics Tags Review + Create

**PROJECT DETAILS**  
Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription \* ⓘ

Resource group \* ⓘ  [Create new](#)

**RESOURCE DETAILS**

Resource name \* ⓘ  Enter the name

Location \* ⓘ

Size ⓘ **A1**  
1 v-core, 3 GB memory  
[Change size](#)

Power BI capacity administrator \* ⓘ  [Select](#)

---

**Review + Create** [Next : Tags >](#)

- **Subscription** - The subscription you would like to create the capacity against.
- **Resource group** - The resource group that contains this new capacity. Pick from an existing resource group, or create another. For more information, see [Azure Resource Manager overview](#).
- **Resource name** - The resource name of the capacity.
- **Location** - The location where Power BI is hosted for your tenant. Your default location is your home region, but you can change the location using [Multi-Geo options](#).
- **Size** - The [A SKU](#) you require. For more information, see [SKU memory and computing power](#).
- **Power BI capacity administrator** - An admin for the capacity.

## Upgrade a capacity to Gen2

You can upgrade a Gen1 capacity to Gen2 in either the Azure UI portal, or the ARM API.

- [Azure UI](#)
- [ARM API](#)

To upgrade a Gen1 capacity to Gen2 in the Azure UI:

1. Select the capacity.
2. From the overview section select **Update to Gen 2**.



3. Select **Yes** to confirm.

 Update to Gen 2  Start  Move  Delete

## Update 'testCapacity' to Embedded Generation 2'

Are you sure you want to update to Embedded Generation 2?

 Yes

No

Status

Paused

Location

The capacity will update automatically in a few seconds.

# AZURE STREAM ANALYTICS SERVICE

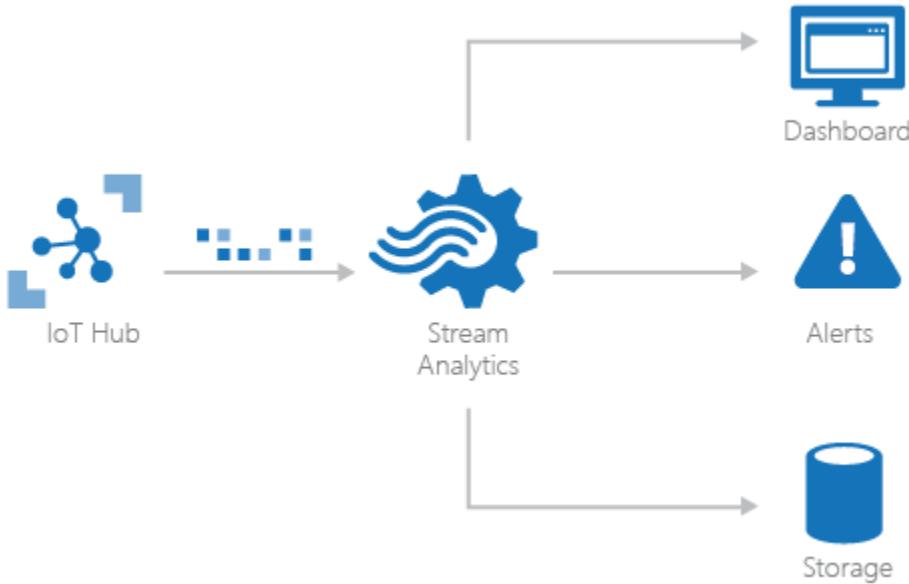


What is azure stream analytics service?



## What is Azure Stream Analytics?

Azure Stream Analytics is a fully managed, serverless engine by Microsoft for real-time analytics. It offers the possibility to perform real-time analytics on multiple streams of data from sources such as sensors, web data sources, social media and other applications.



## When to use Azure Stream Analytics?

You have incoming live streaming data that you want to just store, or report on it with Power BI, or get insights by transforming it, then Azure Stream Analytics might be the service you are looking for. Azure Stream Analytics is a perfect solution if you want a fully managed service where you don't have to worry about any infrastructure setup, and you pay only for what you use.

Azure Stream Analytics use cases:

- real-time dashboarding with Power BI (monitoring purposes)
- store streaming data to make it available to other cloud services for further analysis, logging, reporting etc.
- transform and analyze data in real-time
- trigger workflows on certain conditions (e.g. run Azure Functions from Stream Analytics job)
- send alerts
- make decisions in real-time
- machine learning (e.g. risk analysis, predictive maintenance, fraud detection, predict trends etc.), although for more advanced analytics it has limited usage

Azure Stream Analytics can be used if the input data is in an AVRO, JSON or CSV format and the application logic can be programmed in a query language like SQL. The whole programming in Azure Stream Analytics job is declarative and it doesn't require you to be an expert in programming.

Alternatives for stream processing use cases: Azure Functions, HDInsight with Spark Streaming or Storm, Apache Spark in Azure Databricks

## How to get started with Azure Stream Analytics?

You need an Azure subscription to get started with Azure Stream Analytics and it can be hosted in a few minutes through the Azure portal, PowerShell or Visual Studio. To get started with some real-time analytics you will need to create an Azure Stream Analytics job.

Azure Stream Analytics job is defined by:

- **Input** source of streaming data
- **Query** in a SQL-like language to transform data
- **Output** sink for the results of the data transformations



Key features:

- You can combine data coming from **multiple streams**
- You can use **declarative SQL-based** queries for data transformations
- You can stream the data to **real-time dashboards** with Power BI
- You can **integrate** with Azure IoT Hub

- You only **pay** for streaming units used
- **You don't need to handle** infrastructure
- You will automatically benefit from writing different partitions in **parallel** (increased throughput)
- Your jobs can be visually **monitored**
- **You have recovery capabilities**
- **You can perform operations on data in temporal windows such as tumbling, hopping, sliding and session windows**
- **You have built-in geospatial functions**

Limitation:

- It only supports **SQL** (you are limited to SQL-possible-transformation)
- **Your input data needs to be AVRO, JSON or CSV**
- **You can only use blob storage** to add static data
- **You can only integrate** with Azure services
- **You can't benefit from support dynamic reference data join**
- **There is no automatic scaling (scale job in Azure Portal)**

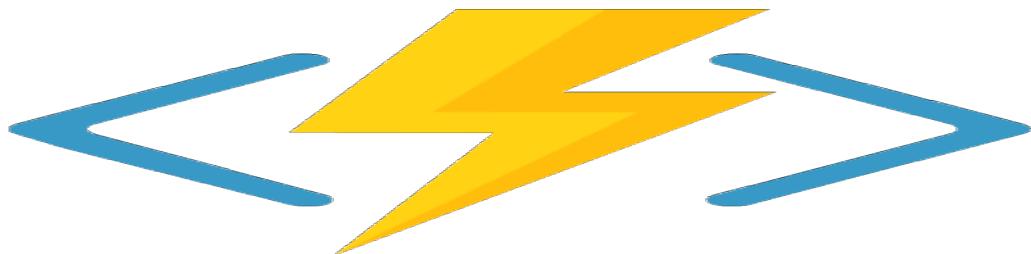
## Alternatives for Streaming Analytics



- Apache Kafka Streaming

Kafka is an open-source product which can run on Azure through HDInsight. It has a real-time streaming functionality Kafka Streaming, yet it will only work if you leverage Kafka as an Event Hub (instead of for example Azure Event Hubs). Additionally, given it's open-source, you are responsible for configuration and maintenance.

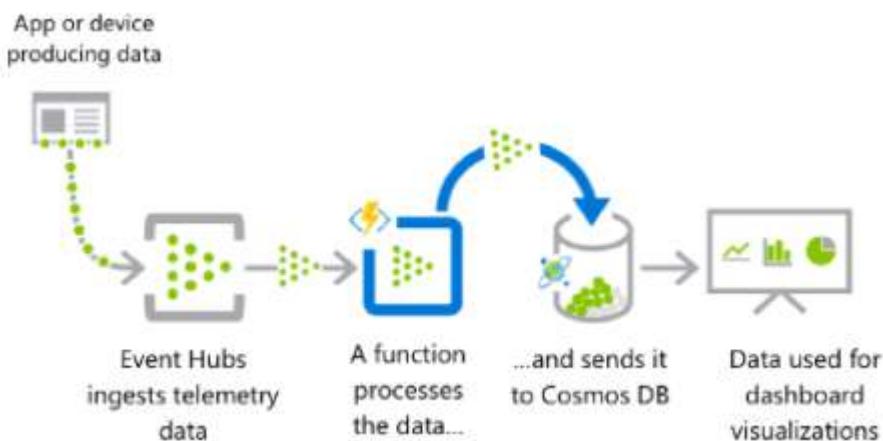
- 



- 

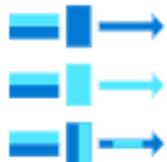
### Azure Functions

Azure Functions is a PaaS serverless service within Azure allowing users to specify functions in Python, .Net or JavaScript. It auto-scales and guarantees high availability and scalability. Through the use of Python, you can apply a broad set of transformations.



## USES:

End-to-end analytics pipeline that is production-ready in minutes with familiar SQL syntax and extensible with JavaScript and C# custom code



Rapid scalability with elastic capacity to build robust streaming data pipelines and analyse millions of events at subsecond latencies



Hybrid architectures for stream processing with the ability to run the same queries in the cloud and on the edge



Enterprise-grade reliability with built-in recovery and built-in machine learning capabilities for advanced scenarios



# Create a Stream Analytics job by using the Azure portal

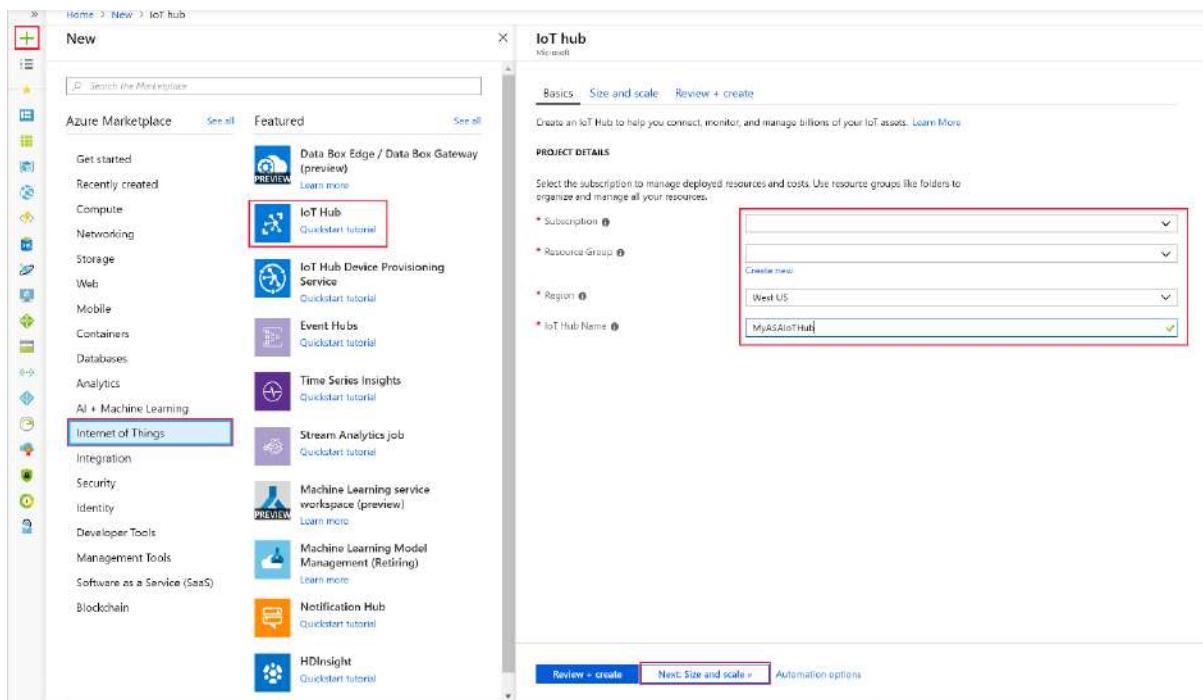
## Prepare the input data

Before defining the Stream Analytics job, you should prepare the input data. The real-time sensor data is ingested to IoT Hub, which later configured as the job input. To prepare the input data required by the job, complete the following steps:

1. Sign in to the [Azure portal](#).
2. Select **Create a resource > Internet of Things > IoT Hub**.
3. In the **IoT Hub** pane, enter the following information:

TABLE 1

Setting	Suggested value	Description
Subscription	<Your subscription>	Select the Azure subscription that you want to use.
Resource group	a <b>quickstart-resourcegroup</b>	Select <b>Create New</b> and enter a new resource-group name for your account.
Region	<Select the region that is closest to your users>	Select a geographic location where you can host your IoT Hub. Use the location that's closest to your users.
IoT Hub Name	MyASAIoTHub	Select a name for your IoT Hub.



5. Select **Next: Set size and scale**.
6. Choose your **Pricing and scale tier**. For this quickstart, select the **F1 - Free** tier if it's still available on your subscription. For more information, see [IoT Hub pricing](#).

**IoT hub**  
Microsoft

**Basics** **Size and scale** **Review + create**

Each IoT Hub is provisioned with a certain number of units in a specific tier. The tier and number of units determine the maximum daily quota of messages that you can send. [Learn more](#)

**SCALE TIER AND UNITS**

\* Pricing and scale tier **S1: Standard tier** [Learn how to choose the right IoT Hub tier for your solution](#)

Number of S1 IoT Hub units **1**

This determines your IoT Hub scale capability and can be changed as your need increases.

Pricing and scale tier <b>S1</b>	Device-to-cloud-messages <b>Enabled</b>
Messages per day <b>400,000</b>	Message routing <b>Enabled</b>
Cost per month <price>	Cloud-to-device commands <b>Enabled</b>
	IoT Edge <b>Enabled</b>
	Device management <b>Enabled</b>

**Advanced Settings**

**Review + create** [Previous: Basics](#) [Automation options](#)

7. Select **Review + create**. Review your IoT Hub information and click **Create**. Your IoT Hub might take a few minutes to create. You can monitor the progress in the **Notifications** pane.
8. In your IoT Hub navigation menu, click **Add** under **IoT devices**. Add a **Device ID** and click **Save**.



9. Once the device is created, open the device from the **IoT devices** list. Copy the **Connection string -- primary key** and save it to a notepad to use later.

Module Identity Name	Connection State	Connection State Last Updated	Last Activity Time
No module identities listed.			

## Create blob storage

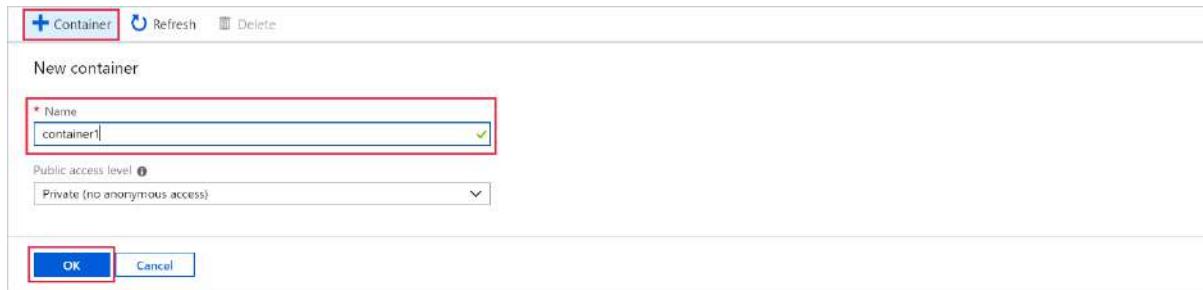
1. From the upper left-hand corner of the Azure portal, select **Create a resource > Storage > Storage account**.
2. In the **Create storage account** pane, enter a storage account name, location, and resource group. Choose the same location and resource group as the IoT Hub you created. Then click **Review + create** to create the account.

The screenshot shows the Azure portal's 'Create storage account' wizard. The left sidebar has 'Storage' selected. The main area shows the 'Storage account - blob, file, table, queue' option highlighted with a red box. The 'Basics' tab is active. The 'Subscription' dropdown shows 'myasauickstartstorage'. The 'Storage account name' field contains 'myasauickstartstorage'. The 'Location' field is set to 'West US'. Other settings like 'Performance' (Standard), 'Account kind' (StorageV2), and 'Replication' (Locally-redundant storage) are also visible. The 'Review + create' button is at the bottom.

3. Once your storage account is created, select the **Blobs** tile on the **Overview** pane.

The screenshot shows the 'myasauickstartstorage' storage account overview page. The 'Overview' tab is selected in the left sidebar. The 'Blobs' tile is highlighted with a red box. Other tiles for 'Files', 'Tables', 'Queues', and 'Monitoring' are also visible. The right side of the screen displays detailed account information like Resource group, Status, Location, Subscription, and Services.

- From the **Blob Service** page, select **Container** and provide a name for your container, such as *container1*. Leave the **Public access level** as **Private (no anonymous access)** and select **OK**.



## Create a Stream Analytics job

- Sign in to the Azure portal.
- Select **Create a resource** in the upper left-hand corner of the Azure portal.
- Select **Analytics > Stream Analytics job** from the results list.
- Fill out the Stream Analytics job page with the following information:

**TABLE 2**

Setting	Suggested value	Description
Job name	MyASAJob	Enter a name to identify your Stream Analytics job. Stream Analytics job name can contain alphanumeric characters, hyphens, and underscores only and it must be between 3 and 63 characters long.
Subscription	< Your subscription>	Select the Azure subscription that you want to use for this job.
Resource group	asaquickstart-resourcegroup	Select the same resource group as your IoT Hub.
Location	<Select the region that is closest to your users>	Select geographic location where you can host your Stream Analytics job. Use the location that's closest to your users for better performance and to reduce the data transfer cost.

Streaming units	1	Streaming units represent the computing resources that are required to execute a job. By default, this value is set to 1. To learn about scaling streaming units, refer to <a href="#">understanding and adjusting streaming units</a> article.
Hosting environment	Cloud	Stream Analytics jobs can be deployed to cloud or edge. Cloud allows you to deploy to Azure Cloud, and Edge allows you to deploy to an IoT Edge device.

The screenshot shows the Azure Stream Analytics job creation interface. On the left, a sidebar lists various service categories. The 'Analytics' category is highlighted with a red box. Within the 'Analytics' section, the 'Stream Analytics job' item is also highlighted with a red box. The main right-hand pane displays the configuration for creating a new job named 'MyASAJob'. It includes fields for subscription, resource group, location, hosting environment (set to 'Cloud'), and streaming units (set to 3). At the bottom right of the configuration pane, there is a 'Create' button.

5.

6. Check the **Pin to dashboard** box to place your job on your dashboard and then select **Create**.
7. You should see a *Deployment in progress...* notification displayed in the top right of your browser window.

## Configure job input

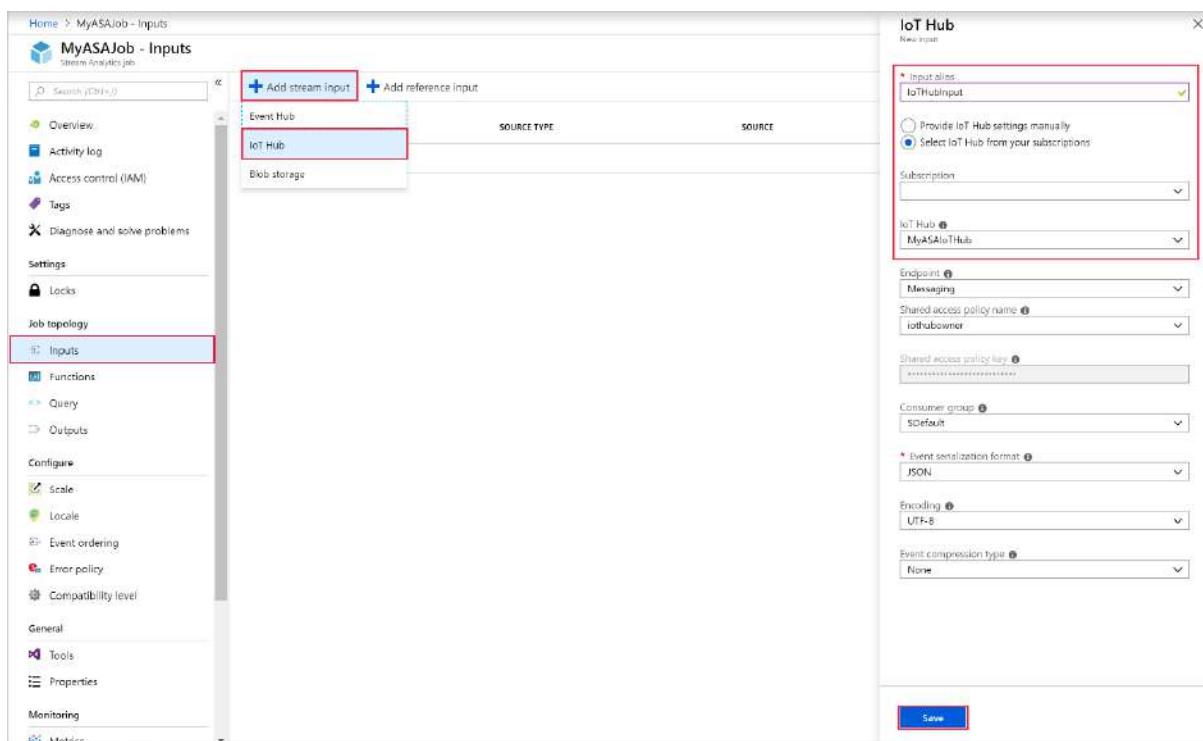
In this section, you will configure an IoT Hub device input to the Stream Analytics job. Use the IoT Hub you created in the previous section of the quickstart.

1. Navigate to your Stream Analytics job.
2. Select **Inputs > Add Stream input > IoT Hub**.
3. Fill out the **IoT Hub** page with the following values:

TABLE 3

Setting	Suggested value	Description
Input alias	IoTHubInput	Enter a name to identify the job's input.
Subscription	< Your subscription>	Select the Azure subscription that has the storage account you created. The storage account can be in the same or in a different subscription. This example assumes that you have created storage account in the same subscription.
IoT Hub	MyASAloTHub	Enter the name of the IoT Hub you created in the previous section.

4. Leave other options to default values and select **Save** to save the settings.



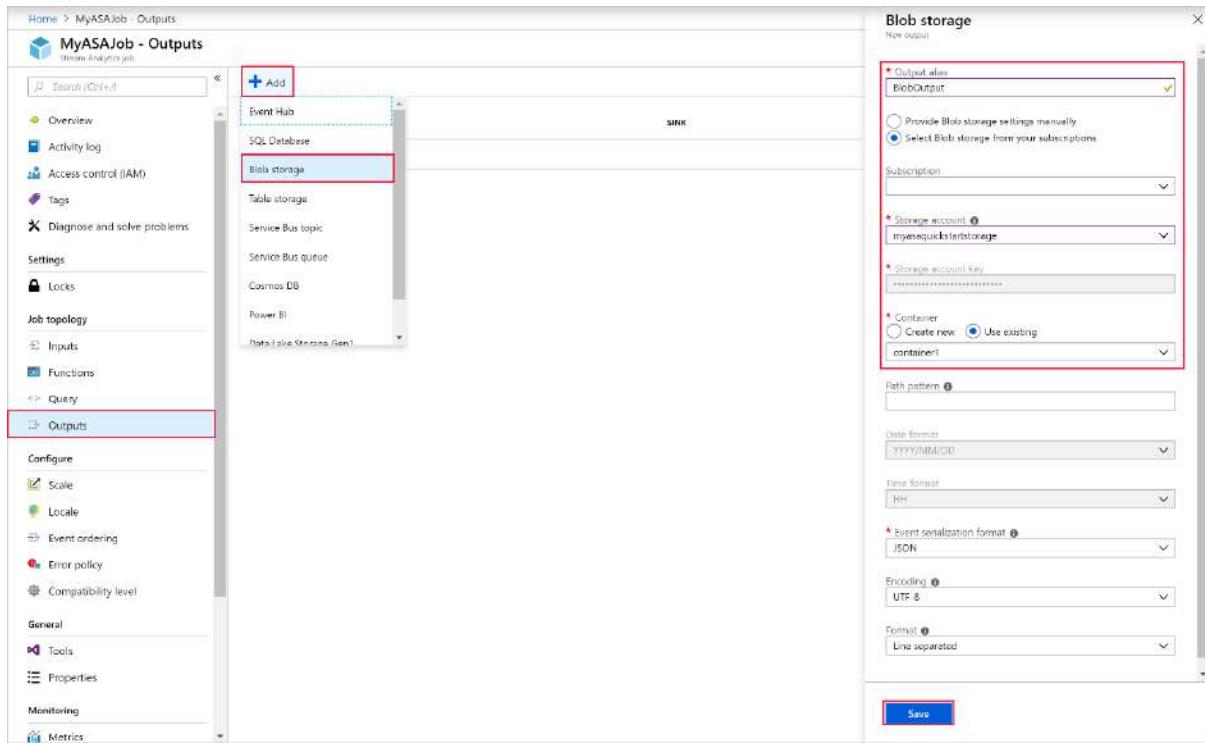
## Configure job output

1. Navigate to the Stream Analytics job that you created earlier.
2. Select **Outputs > Add > Blob storage**.
3. Fill out the **Blob storage** page with the following values:

**TABLE 4**

Setting	Suggested value	Description
Output alias	BlobOutput	Enter a name to identify the job's output.
Subscription	< Your subscription>	Select the Azure subscription that has the storage account you created. The storage account can be in the same or in a different subscription. This example assumes that you have created storage account in the same subscription.
Storage account	asaquickstartstorage	Choose or enter the name of the storage account. Storage account names are automatically detected if they are created in the same subscription.
Container	container1	Select the existing container that you created in your storage account.

4. Leave other options to default values and select **Save** to save the settings.



## Define the transformation query

1. Navigate to the Stream Analytics job that you created earlier.
2. Select **Query** and update the query as follows:

SQLCopy

```
SELECT *
INTO BlobOutput
FROM IoTHubInput
WHERE Temperature > 27
```

3. In this example, the query reads the data from IoT Hub and copies it to a new file in the blob. Select **Save**.

MyASAJob - Query

Save Discard Test

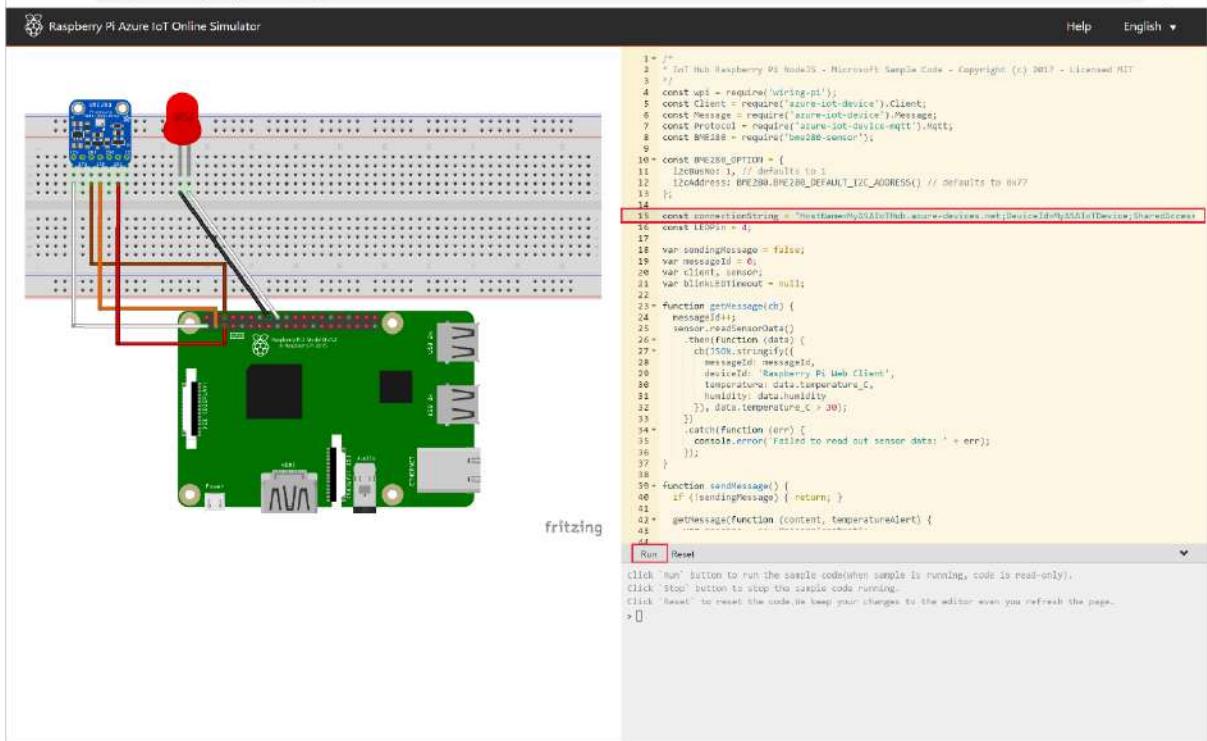
Inputs (1)  
IoTHubInput

Outputs (1)  
BlobOutput

```
1 SELECT
2 *
3 INTO
4 BlobOutput
5 FROM
6 IoTHubInput
7 HAVING temperature > 27
```

## Run the IoT simulator

1. Open the [Raspberry Pi Azure IoT Online Simulator](#).
2. Replace the placeholder in Line 15 with the Azure IoT Hub device connection string you saved in a previous section.
3. Click **Run**. The output should show the sensor data and messages that are being sent to your IoT Hub.



## Start the Stream Analytics job and check the output

1. Return to the job overview page and select **Start**.
2. Under **Start job**, select **Now**, for the **Job output start time** field. Then, select **Start** to start your job.
3. After few minutes, in the portal, find the storage account & the container that you have configured as output for the job. You can now see the output file in the container. The job takes a few minutes to start for the first time, after it is started, it will continue to run as the data arrives.

```

[{"messageId":775,"deviceId":"Raspberry Pi Web Client","temperature":27.56286555465888,"humidity":77}, {"messageId":776,"deviceId":"Raspberry Pi Web Client","temperature":31.76983380970018,"humidity":64}, {"messageId":777,"deviceId":"Raspberry Pi Web Client","temperature":28.173582241045128,"humidity":61}, {"messageId":778,"deviceId":"Raspberry Pi Web Client","temperature":29.78036239344644,"humidity":76}, {"messageId":779,"deviceId":"Raspberry Pi Web Client","temperature":29.537158745343632,"humidity":73}, {"messageId":780,"deviceId":"Raspberry Pi Web Client","temperature":30.11726573108575,"humidity":66}, {"messageId":781,"deviceId":"Raspberry Pi Web Client","temperature":27.237104885232633,"humidity":63}, {"messageId":782,"deviceId":"Raspberry Pi Web Client","temperature":30.541928098456946,"humidity":66}, {"messageId":783,"deviceId":"Raspberry Pi Web Client","temperature":30.46121575922222,"humidity":61}, {"messageId":791,"deviceId":"Raspberry Pi Web Client","temperature":31.687945217662682,"humidity":75}, {"messageId":792,"deviceId":"Raspberry Pi Web Client","temperature":29.066529012500579,"humidity":79}, {"messageId":794,"deviceId":"Raspberry Pi Web Client","temperature":28.859902591436692,"humidity":69}, {"messageId":796,"deviceId":"Raspberry Pi Web Client","temperature":30.78280613043248,"humidity":71}, {"messageId":797,"deviceId":"Raspberry Pi Web Client","temperature":29.466947493481154,"humidity":63}, {"messageId":800,"deviceId":"Raspberry Pi Web Client","temperature":31.1118801842064,"humidity":64}, {"messageId":801,"deviceId":"Raspberry Pi Web Client","temperature":27.136869152806462,"humidity":64}, {"messageId":803,"deviceId":"Raspberry Pi Web Client","temperature":31.197715957953303,"humidity":68}, {"messageId":804,"deviceId":"Raspberry Pi Web Client","temperature":31.612304161725575,"humidity":60}, {"messageId":805,"deviceId":"Raspberry Pi Web Client","temperature":31.733574107399651,"humidity":68}, {"messageId":806,"deviceId":"Raspberry Pi Web Client","temperature":29.535206236689539,"humidity":62}, {"messageId":807,"deviceId":"Raspberry Pi Web Client","temperature":27.860822570967593,"humidity":66}, {"messageId":808,"deviceId":"Raspberry Pi Web Client","temperature":27.775068973736495,"humidity":69}, {"messageId":809,"deviceId":"Raspberry Pi Web Client","temperature":30.5594199223582,"humidity":69.1}, {"messageId":810,"deviceId":"Raspberry Pi Web Client","temperature":30.96622244545026,"humidity":64}, {"messageId":811,"deviceId":"Raspberry Pi Web Client","temperature":29.411526757613782,"humidity":69}, {"messageId":812,"deviceId":"Raspberry Pi Web Client","temperature":28.086333098844142,"humidity":70}, {"messageId":822,"deviceId":"Raspberry Pi Web Client","temperature":30.849464815252702,"humidity":64}, {"messageId":823,"deviceId":"Raspberry Pi Web Client","temperature":27.63494418804433,"humidity":62}, {"messageId":827,"deviceId":"Raspberry Pi Web Client","temperature":31.171863047724845,"humidity":67}, {"messageId":831,"deviceId":"Raspberry Pi Web Client","temperature":31.748161497954346,"humidity":64}

```

## Clean up resources

When no longer needed, delete the resource group, the Stream Analytics job, and all related resources. Deleting the job avoids billing the streaming units consumed by the job. If you're planning to use the job in future, you can stop it and restart it later when you need. If you are not going to continue to use this job, delete all resources created by this quickstart by using the following steps:

1. From the left-hand menu in the Azure portal, select **Resource groups** and then select the name of the resource you created.
2. On your resource group page, select **Delete**, type the name of the resource to delete in the text box, and then select **Delete**.

# AZURE SYNAPSE ANALYTICS SERVICE

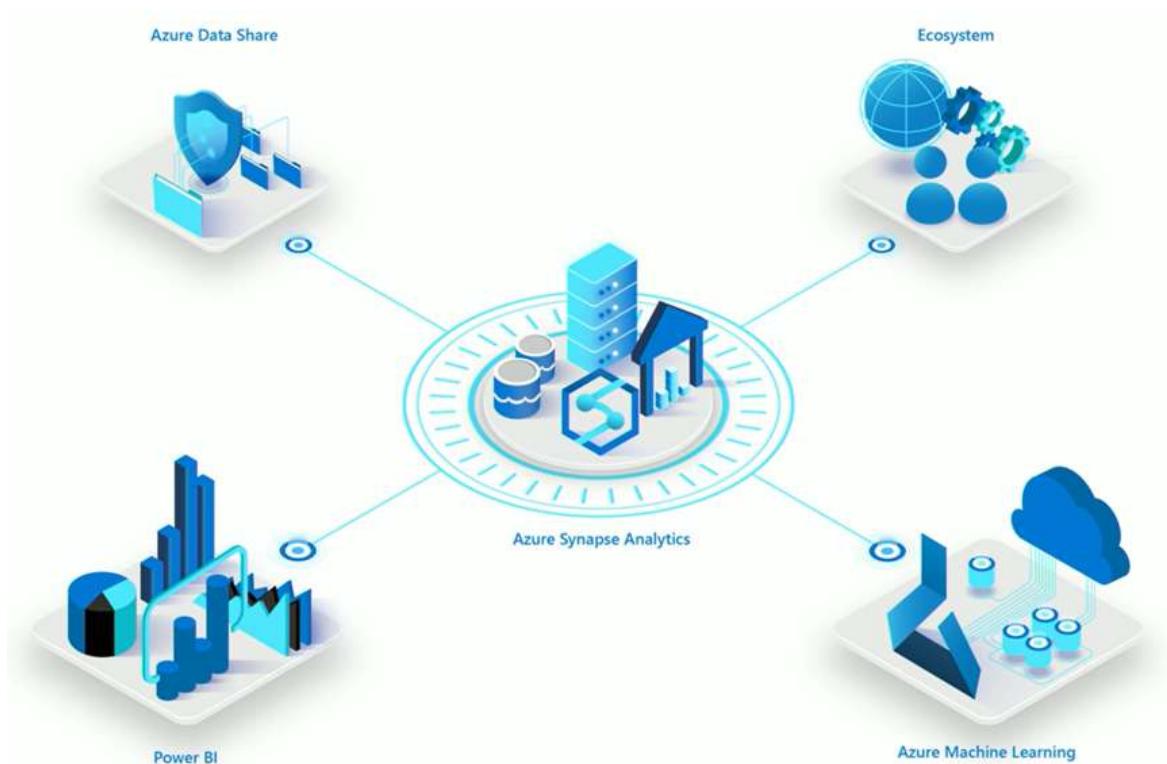


**Azure Synapse Analytics**

## What is Azure Synapse ?

Azure Synapse is a limitless analytics service that brings together enterprise data warehousing and Big Data analytics. It gives you the freedom to query data on your terms, using either server-less or provisioned resources at scale. Azure Synapse brings these two worlds together with a unified experience to ingest, prepare, manage, and serve data for immediate BI and machine learning needs.

Architecture and Design of a data warehouse depends on many factors. A dynamic and well performing DW should guaranty the data validity, concurrency, low latency and capability to integrate with other systems.



Azure Synapse provides you the platform to build and manage a modern DW with limitless analytics service that brings together enterprise data warehousing and Big Data analytics. It gives you the freedom to query data on your terms, using either server less on-demand or provisioned resources at scale.

## USES:

### Limitless scale

Deliver insights from all your data, across data warehouses and big data analytics systems, with blazing speed.



### Powerful insights

Expand discovery of insights from all your data and apply machine learning models to all your intelligent apps.



### Unified experience

Significantly reduce project development time with a unified experience for developing end-to-end analytics solutions.



# **Key Service Capabilities**

## **Unified analytics platform**

Perform data integration, data exploration, data warehousing, big data analytics and machine learning tasks from a single, unified environment.

## **Serverless and dedicated options**

Support both data lake and data warehouse use cases and choose the most cost-effective pricing option for each workload.

## **Enterprise data warehousing**

Build your mission-critical data warehouse on the proven foundation of the industry's top-performing SQL engine.

## **Data lake exploration**

Bring together relational and nonrelational data and easily query files in the data lake with the same service you use to build data warehousing solutions.

## **Code-free hybrid data integration**

Build ETL/ELT processes in a code-free visual environment to easily ingest data from more than 95 native connectors.

## **Deeply integrated Apache Spark and SQL engines**

Enhance collaboration among data professionals working on advanced analytics solutions. Easily use T-SQL queries on both your data warehouse and Spark engines.

## **Log and telemetry analytics**

Use industry-leading text-indexing technology to gain insights from time-series, log and telemetry data with the Azure Synapse data explorer distributed query engine.

## Choice of language

Use your preferred language, including T-SQL, KQL, Python, Scala, Spark SQL and .Net—whether you use serverless or dedicated resources.

## Integrated AI and BI

Complete your end-to-end analytics solution with deep integration of Azure Machine Learning, Azure Cognitive Services and Power BI.

## Cloud-native HTAP

Get insights from real-time transactional data stored in operational databases, such as Azure Cosmos DB, with a single click.

## Accelerate data warehouse migration with Azure Synapse Pathway

Automate mandatory and critical data warehouse migration steps with a point-and-click solution that scans your source system, produces an inventory report and translates existing code in minutes—not weeks or months. No manual rewriting needed—get more than 100,000 lines of SQL code translated in minutes.



## Get Started with Azure Synapse Analytics

Follow the steps *in order* as shown below and you'll take a tour through many of the capabilities and learn how to exercise its core features.

- **STEP 1 - Create and setup a Synapse workspace**
- **STEP 2 - Analyze using a serverless SQL pool**
- **STEP 3 - Analyze using a Data Explorer pool**
- **STEP 4 - Analyze using Apache Spark**
- **STEP 5 - Analyze using a dedicated SQL pool**
- **STEP 6 - Analyze data in a storage account**
- **STEP 7 - Orchestrate with pipelines**
- **STEP 8 - Visualize data with Power BI**
- **STEP 9 - Monitor activities**
- **STEP 10 - Explore the Knowledge center**
- **STEP 11 - Add an administrator**

## Create Azure Synapse Workspace

- **SQL on demand** - This is a serverless service, allowing you to do light data explorations in Data Lake. It is provisioned automatically when creating a workspace and users have no control over it.
- **SQL pool** - This is a Synapse database, based on a multi-node cluster, which is also formerly known as SQL DW database. This component is optional and may or may not exist, depending on your Data Warehouse design.
- **Spark pool** - This is a Spark database, based on a multi-node cluster. This is also an optional component.

It is worth mentioning that every Synapse Workspace requires at least one Data Lake account associated with it, which is specified during its creation. Here are the steps to create a Synapse Workspace:

1. Log into the Azure portal, create a new resource and specify the Azure Synapse Analytics (workspaces preview) type.
2. Provide the resource group, workspace name, region, Data Lake storage account and file system name within that account. Alternatively, you can create storage and file system accounts if they do not exist. Here is a sample screenshot:

Microsoft Azure Search resources, services, and docs (G)

Home > New > Azure Synapse Analytics (workspaces preview) >

## Create Synapse workspace

Basics Security + networking Tags Summary

Create a Synapse workspace to develop an enterprise analytics solution in just a few clicks.

**Project details**

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all of your resources.

Subscription \* Visual Studio Professional with MSDN

Resource group \* [REDACTED] [Create new](#)

**Workspace details**

Name your workspace, select a location, and choose a primary Data Lake Storage Gen2 file system to serve as the default location for logs and job output.

Workspace name \* myws

Region \* East US 2

Select Data Lake Storage Gen2 \* From subscription  Manually via URL

Account name \* [REDACTED] [Create new](#)

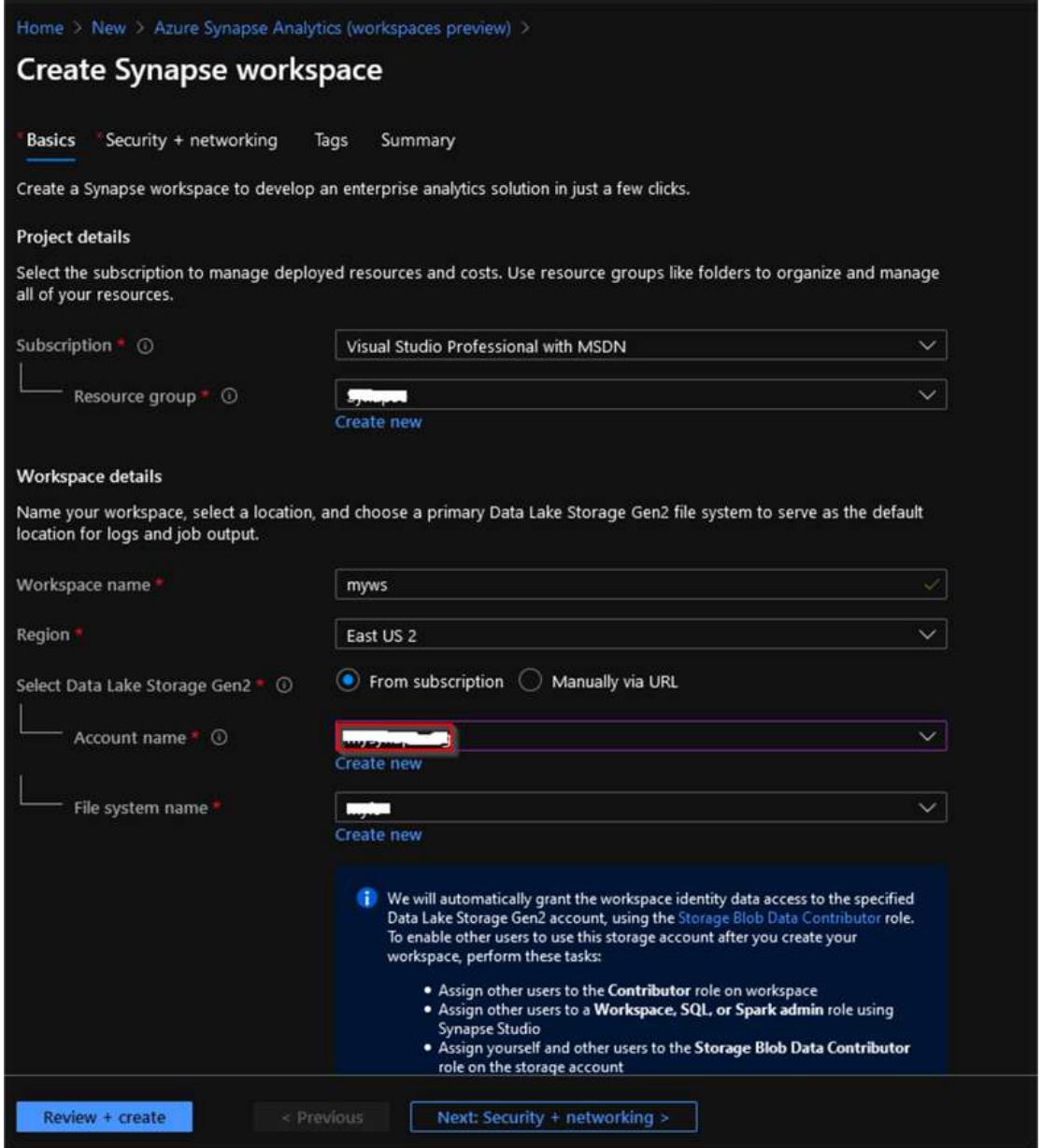
File system name \* [REDACTED] [Create new](#)

**Information**

We will automatically grant the workspace identity data access to the specified Data Lake Storage Gen2 account, using the **Storage Blob Data Contributor** role. To enable other users to use this storage account after you create your workspace, perform these tasks:

- Assign other users to the **Contributor** role on workspace
- Assign other users to a **Workspace, SQL, or Spark admin** role using Synapse Studio
- Assign yourself and other users to the **Storage Blob Data Contributor** role on the storage account

[Review + create](#) [< Previous](#) [Next: Security + networking >](#)

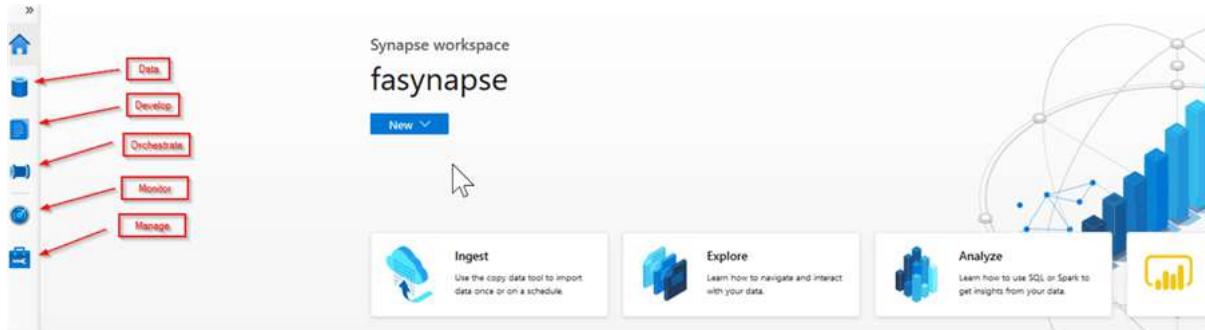


The workspace provisioning may take a few minutes, after which a Synapse Workspace home page opens, as follows:

The screenshot shows the Azure Synapse workspace overview page. On the left, there's a navigation sidebar with sections like Home, Overview, Activity log, Access control (IAM), Tags, Settings, Synapse resources, Security, Monitoring, and Support + troubleshooting. Under Synapse resources, 'SQL pools' and 'Apache Spark pools' are listed. The main central area displays workspace details such as Resource group (Synapse), Status (Succeeded), Location (East US 2), Subscription (Visual Studio Professional with MSDN), and various connection details. Below this is a table titled 'Available resources' showing 'SQL pools' (No pools provisioned) and 'Apache Spark pools' (1 pool named 'fasynapsepool'). The 'Apache Spark pools' section is also highlighted with a red box. At the top right, there's a 'Launch Synapse Studio' button, which is also highlighted with a red box.

The central panel workspace contains a list of your SQL and Spark pools. You can create pools from either this screen, or from the Synapse Studio, which I will describe a bit later. Let's launch Synapse Studio using top-right button shown above.

Here's the Synapse Studio home screen:

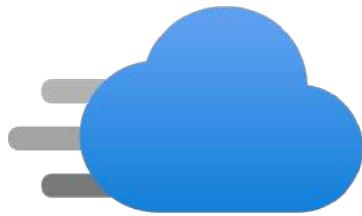


The left panel contains the main commands, as described below:

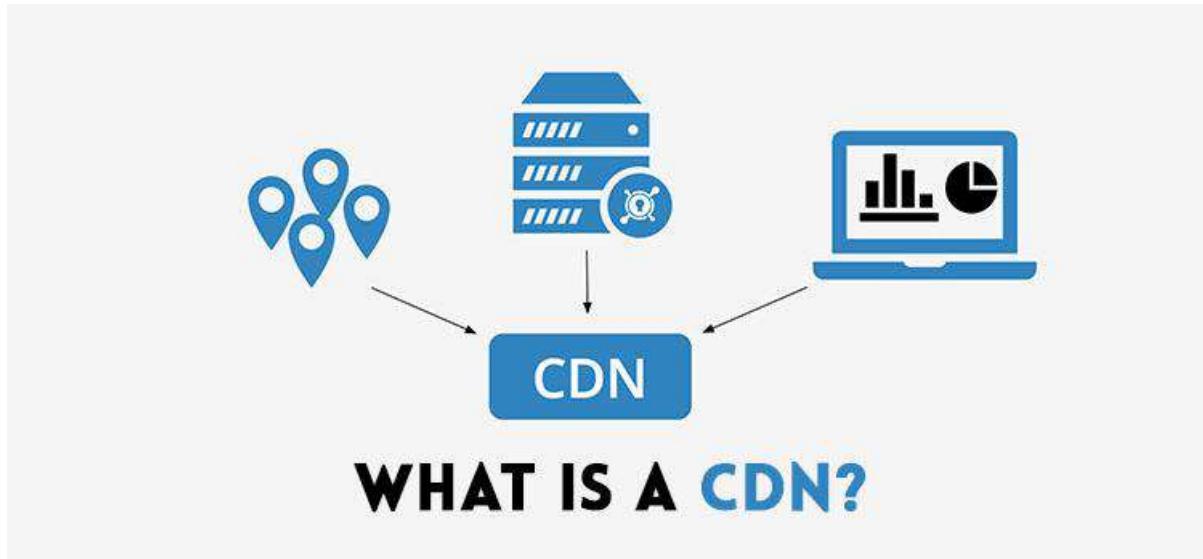
- **Data** - This command allows browsing available databases and Data Lake accounts.
- **Develop** - This command allows creating SQL scripts and Spark notebooks.
- **Orchestrate** - This command allows building ETL pipelines.
- **Monitor** - This command allows monitoring pipeline executions.
- **Manage** - This command allows managing pools, linked services, etc.

This concludes the overview of Azure Synapse Analytics. We will dive into the Azure Synapse Analytics exciting features and learn how to apply them for solving real-life analytics problems in upcoming tips.

# AZURE CONTENT DELIVERY NETWORK(CDN)



## What is Azure Content Delivery Network?



A content delivery network (CDN) is a distributed network of servers that can efficiently deliver web content to users. CDNs store cached content on edge servers in point-of-presence (POP) locations that are close to end users, to minimize latency.

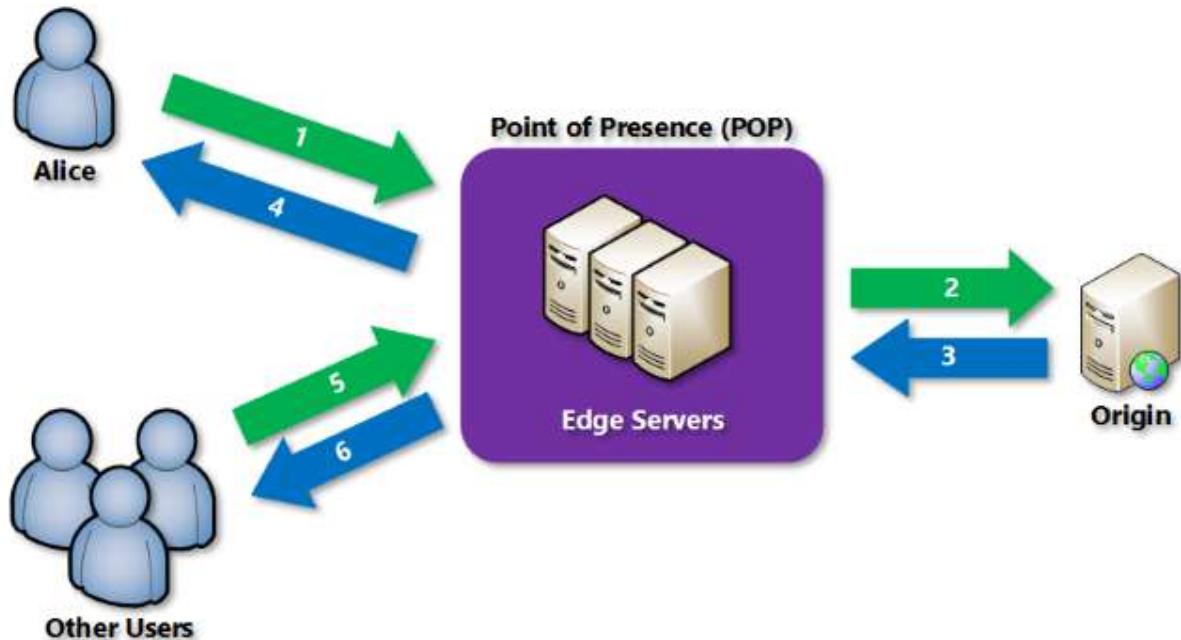
Azure Content Delivery Network (CDN) offers developers a global solution for rapidly delivering high-bandwidth content to users by caching their content at strategically placed physical nodes across the world. Azure CDN can also accelerate dynamic content, which cannot be cached, by leveraging various network optimizations using CDN POPs. For example, route optimization to bypass Border Gateway Protocol (BGP).

The benefits of using Azure CDN to deliver web site assets include:

- Better performance and improved user experience for end users, especially when using applications in which multiple round-trips are required to load content.
- Large scaling to better handle instantaneous high loads, such as the start of a product launch event.

- Distribution of user requests and serving of content directly from edge servers so that less traffic is sent to the origin server.

## How it works?



1. A user (Alice) requests a file (also called an asset) by using a URL with a special domain name, such as <endpoint name>.azureedge.net. This name can be an endpoint hostname or a custom domain. The DNS routes the request to the best performing POP location, which is usually the POP that is geographically closest to the user.
2. If no edge servers in the POP have the file in their cache, the POP requests the file from the origin server. The origin server can be an Azure Web App, Azure Cloud Service, Azure Storage account, or any publicly accessible web server.
3. The origin server returns the file to an edge server in the POP.
4. An edge server in the POP caches the file and returns the file to the original requestor (Alice). The file remains cached on the edge server in the POP until the time-to-live (TTL) specified by its HTTP headers expires. If the origin server didn't specify a TTL, the default TTL is seven days.
5. Additional users can then request the same file by using the same URL that Alice used, and can also be directed to the same POP.
6. If the TTL for the file hasn't expired, the POP edge server returns the file directly from the cache. This process results in a faster, more responsive user experience.

## GETTING STARTED WITH AZURE CDN:

Caching is one of the ways for performance improvement. Windows Azure uses caching to increase the speed of cloud services. Content Delivery Network (CDN) puts stuff like blobs and other static content in a cache. The process involves placing the data at strategically chosen locations and caching it. As a result, it provides maximum bandwidth for its delivery to users. Let's assume an application's source is far away from the end user and many tours are taken over the internet to fetch data; the CDN offers a very competent solution to improve performance in this case. Additionally, it scales the instant high load in a very efficient manner.

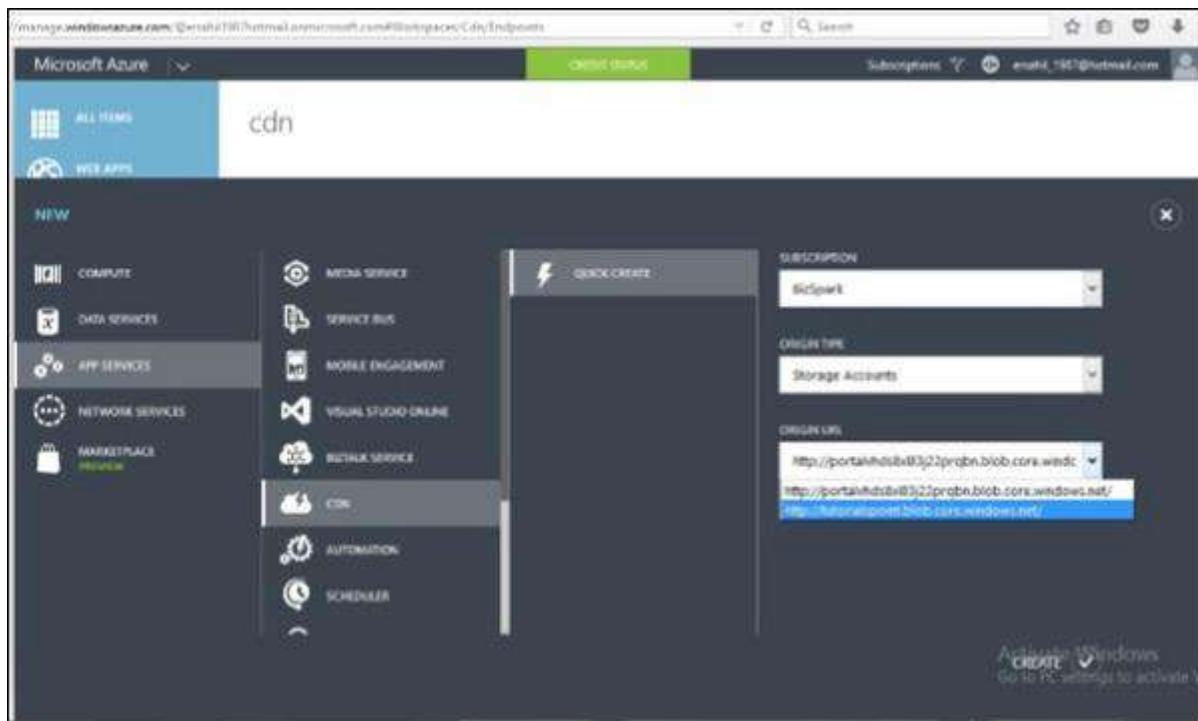
### Create a CDN:

**Step 1** – Login in to your Azure Management Portal.

**Step 2** – Click on 'New' at bottom left corner.

**Step 3** – Select 'APP Services' then 'CDN'.

**Step 4** – Click on 'Quick Create'. The following screen will come up.



You will see three fields in the pop up –

- **Subscription** – There will be a list of subscriptions you have subscribed to and you can choose from one of them. In this demo, only one option was there in the subscription dropdown, which was 'BizSpark', the current subscription.

- **Origin Type** – This dropdown will ask to select an origin type. The integrated service will have an option of Web Apps, Cloud Services, Storage and Media Services.
- **Origin URL** – This will show the URLs based on the chosen origin type in the dropdown.

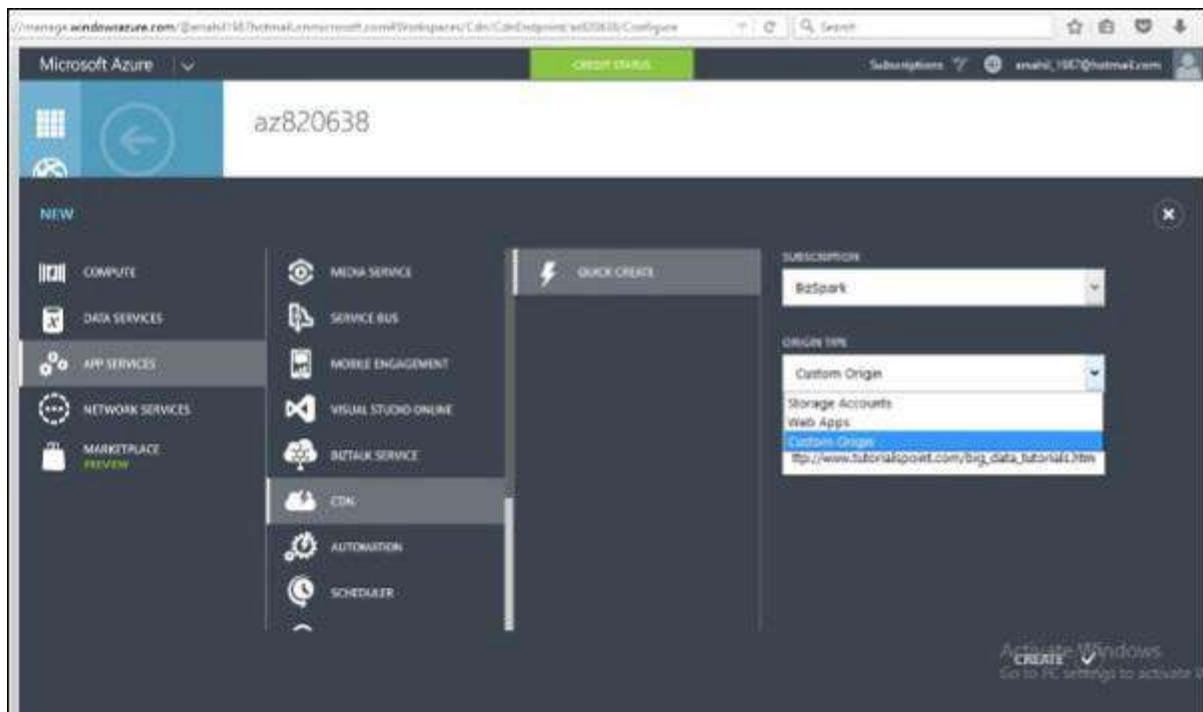
**Step 5** – Choose one of the options from each dropdown as needed and click ‘Create’. CDN endpoint is created as shown in the following image.

NAME	STATUS	SUBSCRIPTION	URL	ORIGIN
ABD963	✓ Enabled	Bharat	http://www.abd963.com	http://www.abd963.com

## Create CDN for Custom Origin Links

In June 2015, CDN was updated with one more feature where users can specify a custom origin. Earlier only Azure services could be linked to CDN, but now any website can be linked to it using this service.

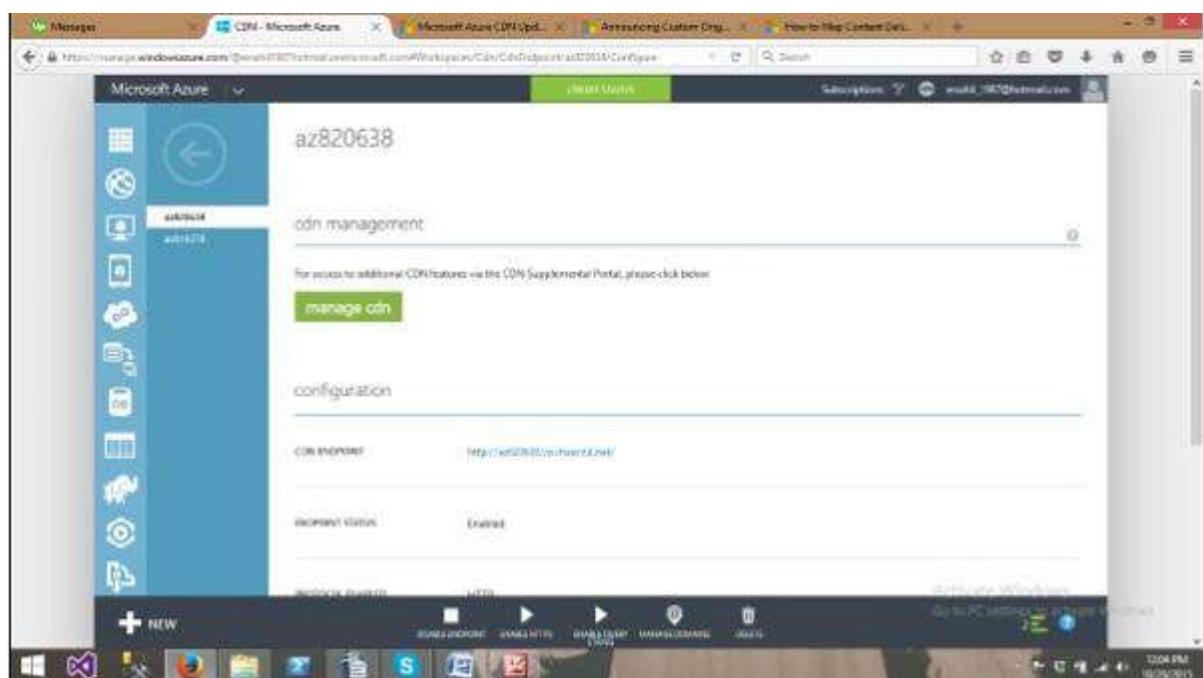
When we are create a CDN service, in the ‘Origin Type’ dropdown, there is an option ‘Custom Origin’ as shown in the following image, and then you can specify the link in the URL field.



## MANAGE CDN:

**Step 1** – Click on the Name of the CDN you want to manage in the list displayed in **CDN services**.

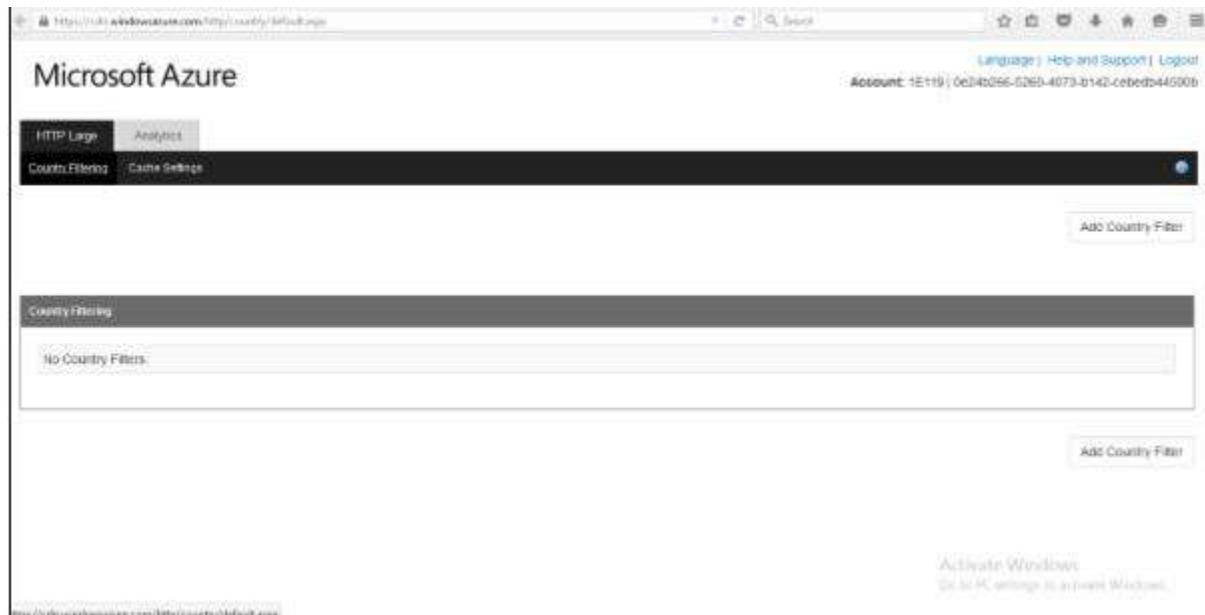
**Step 2** – Click on ‘manage cdn’.



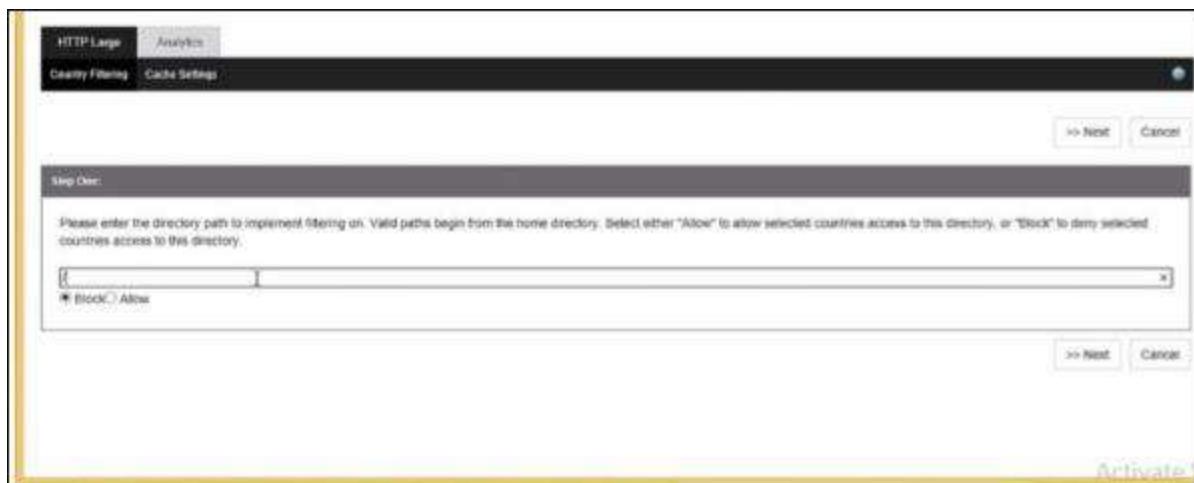
**Country filtering** – You can allow/lock your website in specified countries. This is going to protect your data for better.

**Step 3** – When you click on ‘manage cdn’ you will be taken to the following page in a new tab of your browser.

**Step 4** – Click on ‘Country Filtering’ from menu items at the top of screen. Click on ‘Add Country Filter’ button as shown in the following image.



**Step 5** – Specify the directory and select Allow/block.



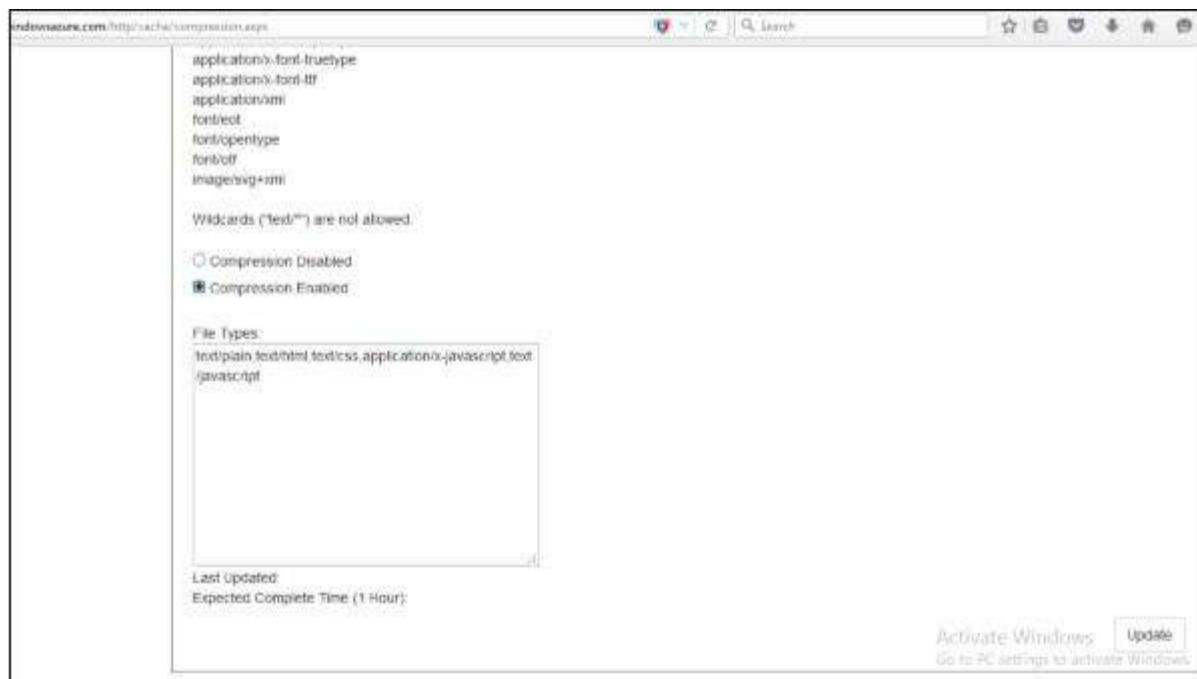
**Step 6** – Select the country in the next screen and you are done.



**Compression** – It allows files to be compressed. You can enable/disable compression. Also you can specify the file type.

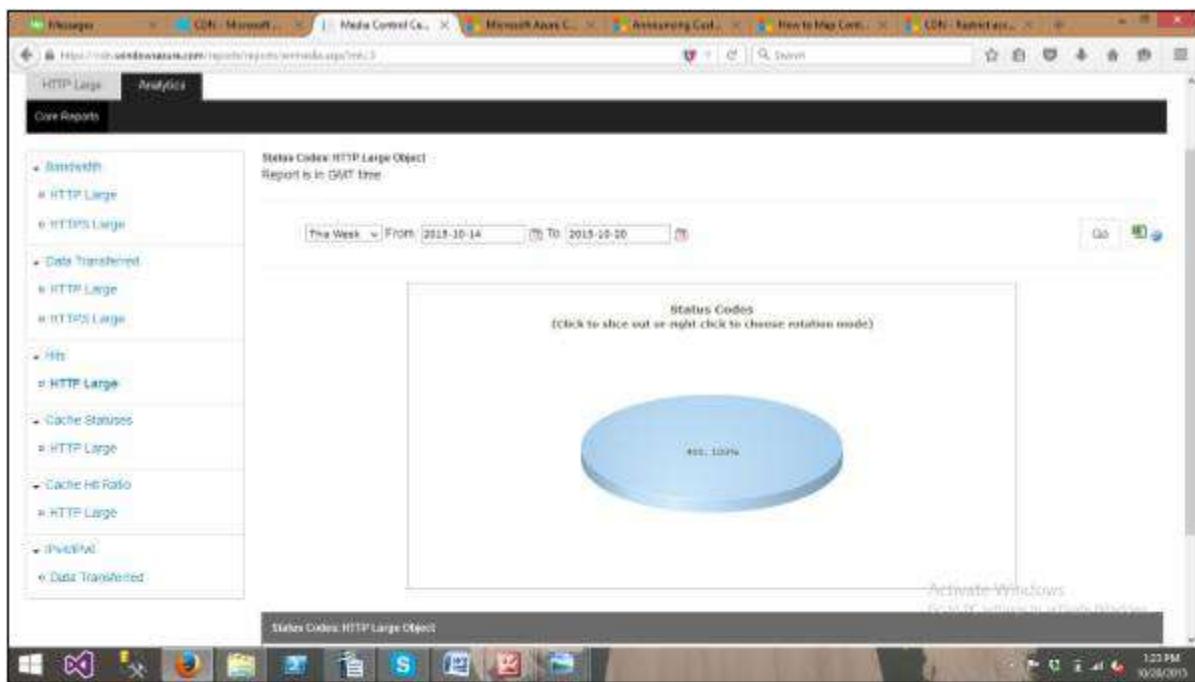
**Step 7** – Click on ‘Cache Setting’ and scroll down to the bottom of the page.

**Step 8** – Select ‘Compression Enabled’ and click ‘Update’ button. By default, compression is disabled.



**Analytics** – You can see very useful figures in this section. For example, number of overall hits or in a specific geographic region. The report will also show how many times requests are served from CDN endpoints and how many of them are going back to the original server.

**Step 9** – Click on ‘Analytics’ in menu items at the top of the page. You will see a list of all the reports in the left panel as shown in the following image.

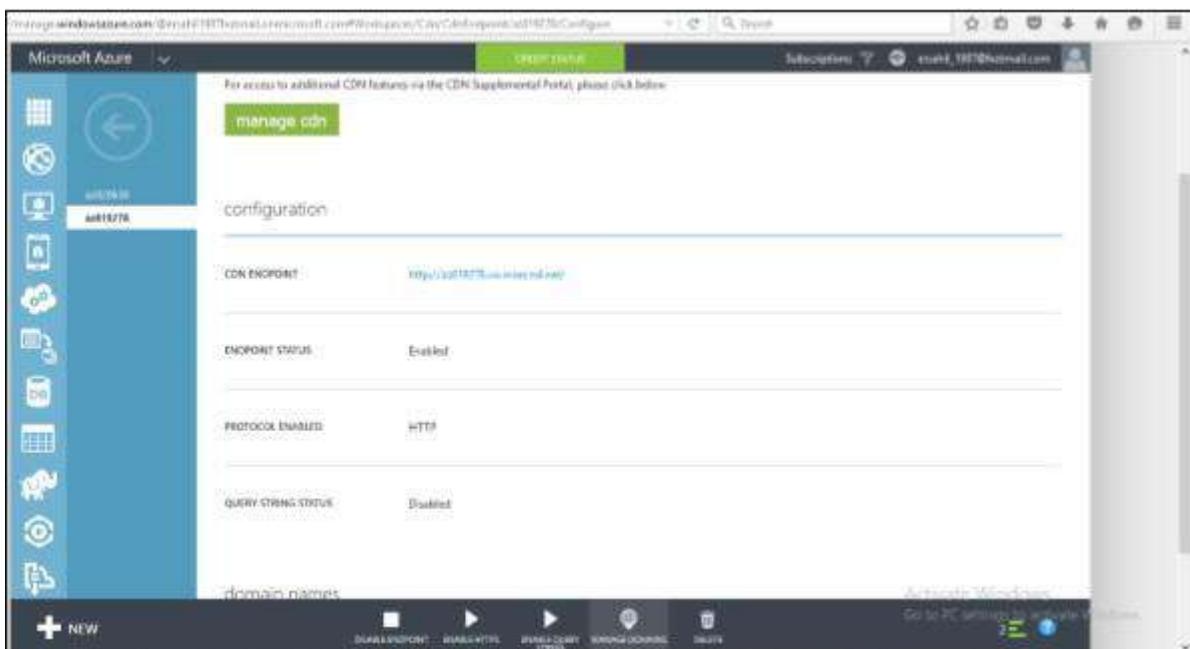


**Step 10** – Additionally, you can download the report as an excel file by clicking on the excel icon at the top right corner.

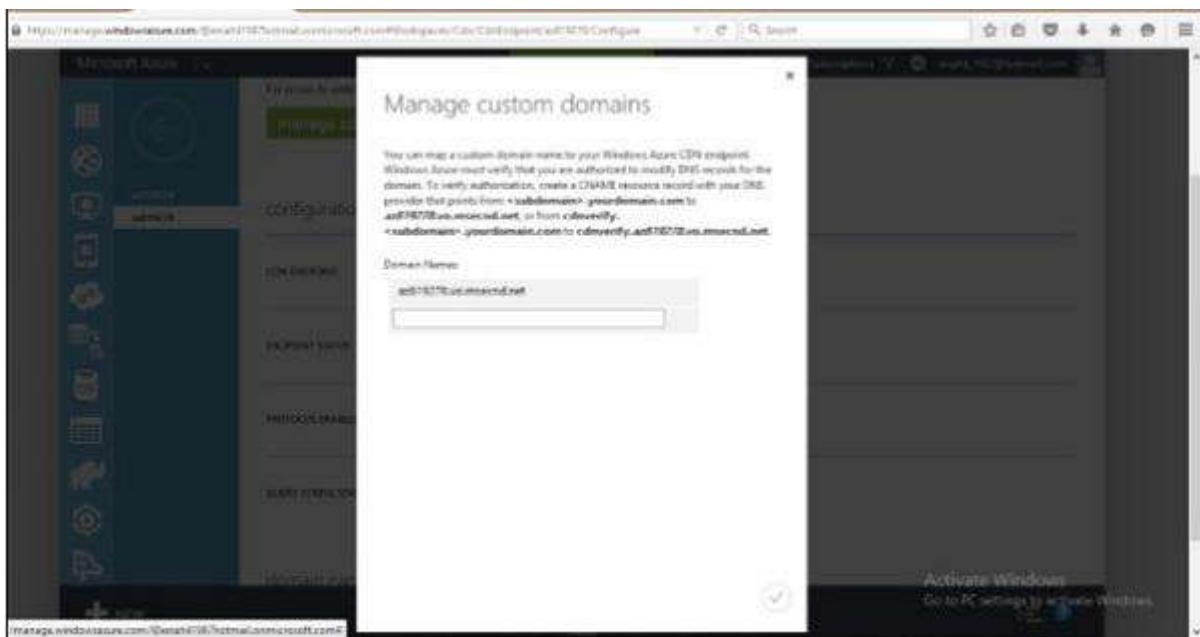
## MAP A CUSTOM DOMAIN NAME:

You might want to use a custom domain name instead of CDN endpoint that is autogenerated by Azure service. Windows Azure has provided a new feature that allows you to map a custom domain name to his application's CDN endpoint.

**Step 1** – Click on ‘Manage Domain’ Button on the bottom horizontal menu.



**Step 2** – Enter the custom URL in the text box and its done.



# AZURE COMMUNICATION SERVICES



## What is Azure Communication Services?

Azure Communication Services are cloud-based services with REST APIs and client library SDKs available to help you integrate communication into your applications. You can add communication features to your applications without being an expert in communication technologies such as media encoding and real-time networking. This functionality is also supported in Azure for government.

Azure Communication Services supports various communication formats:

1. Voice and Video Calling
2. Rich Text Chat
3. SMS

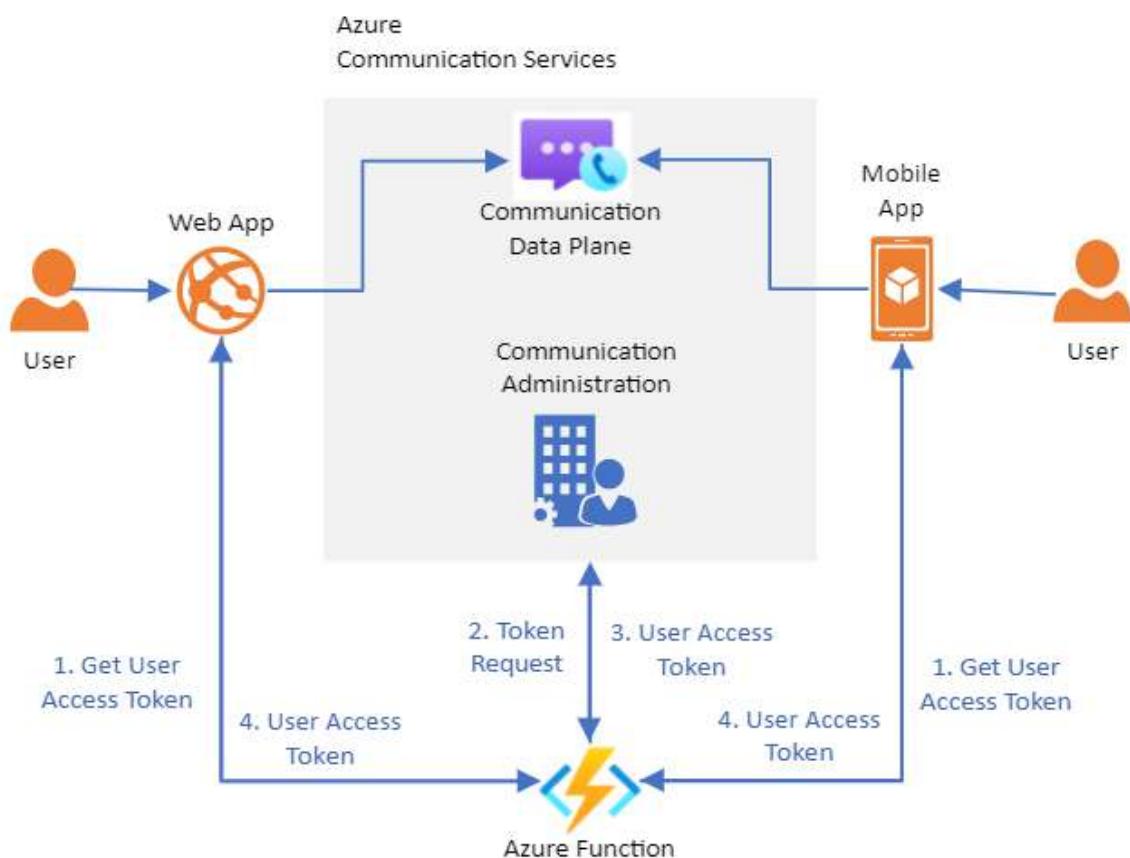
You can connect custom client endpoints, custom services, and the publicly switched telephony network (PSTN) to your communications application. You can acquire phone numbers directly through Azure Communication Services REST APIs, SDKs, or the Azure portal; and use these numbers for SMS or calling applications. Azure Communication Services direct routing allows you to use SIP and session border controllers to connect your own PSTN carriers and bring your own phone numbers.

Scenarios for Azure Communication Services include:

- **Business to Consumer (B2C).** A business' employees and services interact with consumers using voice, video, and rich text chat in a custom browser or mobile application. An organization can send and receive SMS messages, or operate an interactive voice response system (IVR) using a phone number you acquire through Azure. Integration with Microsoft Teams can be used to connect consumers to Teams meetings hosted by employees; ideal for remote healthcare, banking, and product support scenarios where employees might already be familiar with Teams.

- **Consumer to Consumer (C2C).** Build engaging social spaces for consumer-to-consumer interaction with voice, video, and rich text chat. Any type of user interface can be built on Azure Communication Services SDKs, or use complete application samples and an open-source UI toolkit to help you get started quickly.

## BASIC STRUCTURE OF AZURE COMMUNICATION SERVICES



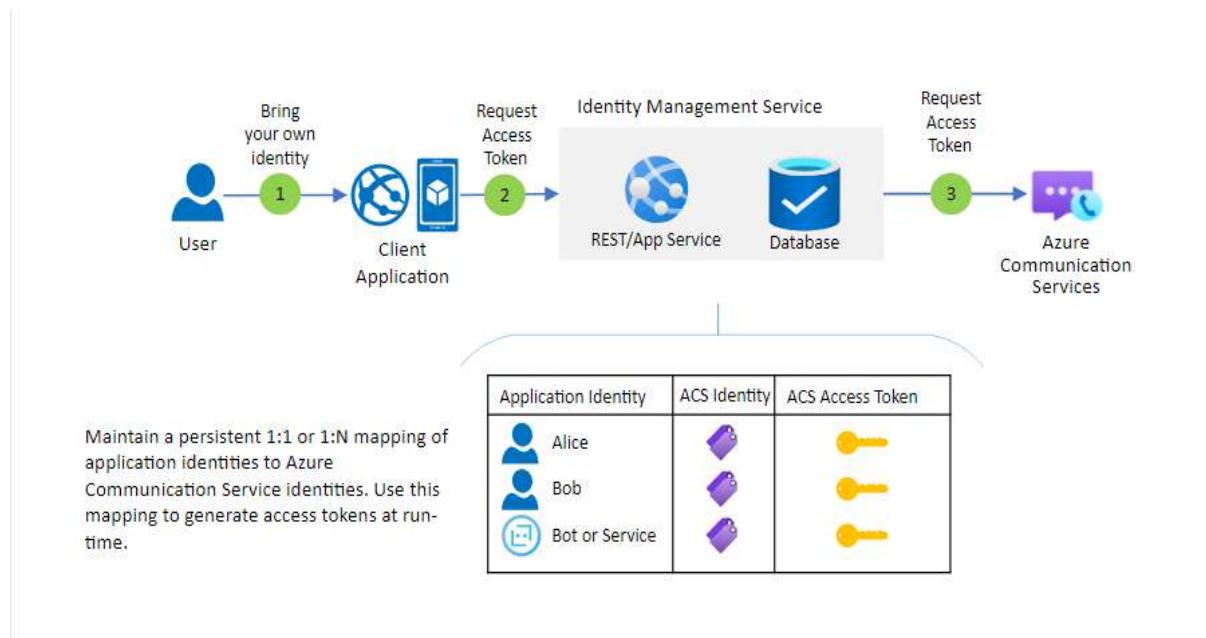
## Client and Server Architecture:

1. **Client Application.** This website or native application is leveraged by end-users to communicate. Azure Communication Services provides SDK client libraries for multiple browsers and application platforms. In addition to our core SDKs, a UI Library is available to accelerate browser app development.
2. **Identity Management Service.** This service capability you build to map users and other concepts in your business logic to Azure Communication Services and also to create tokens for those users when required.

3. **Call Management Service.** This service capability you build to manage and monitor voice and video calls. This service can create calls, invite users, call phone numbers, play audio, listen to DMTF tones and leverage many other call features through the Calling Automation SDK and REST APIs.

## User access management

Azure Communication Services clients must present user access tokens to access Communication Services resources securely. User access tokens should be generated and managed by a trusted service due to the sensitive nature of the token and the connection string or managed identity necessary to generate them. Failure to properly manage access tokens can result in additional charges due to misuse of resources.



### Dataflows:

1. The user starts the client application. The design of this application and user authentication scheme is in your control.
2. The client application contacts your identity management service. The identity management service maintains a mapping between your users and other addressable objects (for example services or bots) to Azure Communication Service identities.
3. The identity management service creates a user access token for the applicable identity. If no Azure Communication Services identity has been allocated the past, a new identity is created.

## AZURE MEDIA SERVICE



It is an extensible cloud-based platform that enables developers to build scalable media management and delivery applications. For example - if we want to develop an app like DailyMotion, then we can do so by using Microsoft Azure media services.

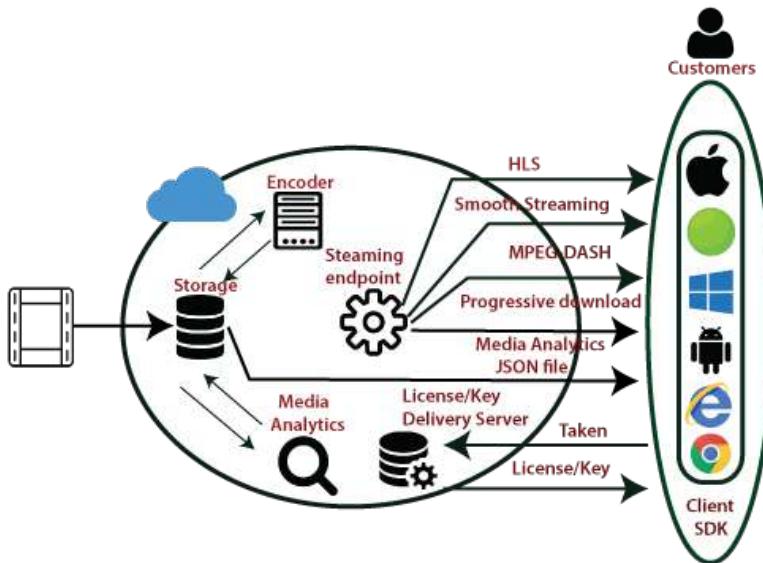
Azure media services are based on REST APIs that enable us to securely upload, store, encode, and wrap video or audio content for both on-demand and live stream delivery to various clients. Those clients can be TV, PC, and mobile devices also.

## Media Services Concepts:

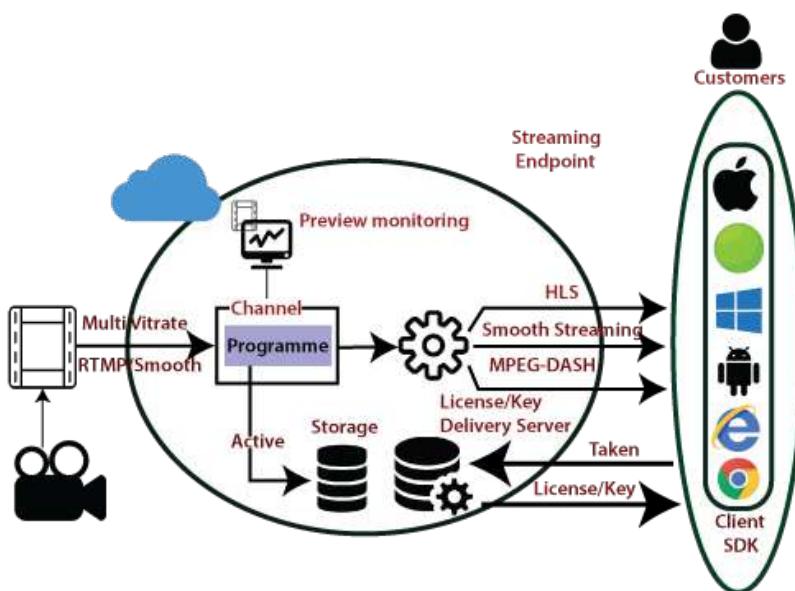
- **Assets:** An Asset contains digital files and the metadata about these files. These files can be audio, video or image, etc.
- **AssetFile:** It contains metadata about the media file.
- **AccessPolicy:** It defines the permission and duration of access to an asset.
- **Locators:** It provides an entry point to access the files contained in an asset.
- **Job:** It is used to process one audio/video presentation.
- **Channels:** It is responsible for processing live streaming content. It provides an input endpoint that is provided to a live transcoder.
- **Program:** It enables us to control the publishing and storage of segments in a live stream.
- **Streaming endpoint:** It represents a streaming service that delivers content.

# The Architecture of Media Service

- **Delivering on-demand:** In this case, first, we will upload a high-quality media file into an asset, and then we encode it to a set of adaptive bit that reads MP4 files. After that, we configure the asset delivery policy. Asset delivery policy tells Media services how we want our assets to be delivered using which protocol. Now, we will publish an asset by creating an on-demand locator and stream the published content.



- **Live-Streaming:** We can broadcast live content using various live streaming protocols. We might go to encode our stream into an adoptive bit read stream. We can preview our live stream also. Finally, we can deliver the content through common streaming protocols such as Smooth, HLS, etc.



# AZURE SEARCH

Azure Search is a cloud Search as a Service that enables us to add a robust search experience to our applications using a simple REST API or .NET SDK, without managing a search infrastructure.

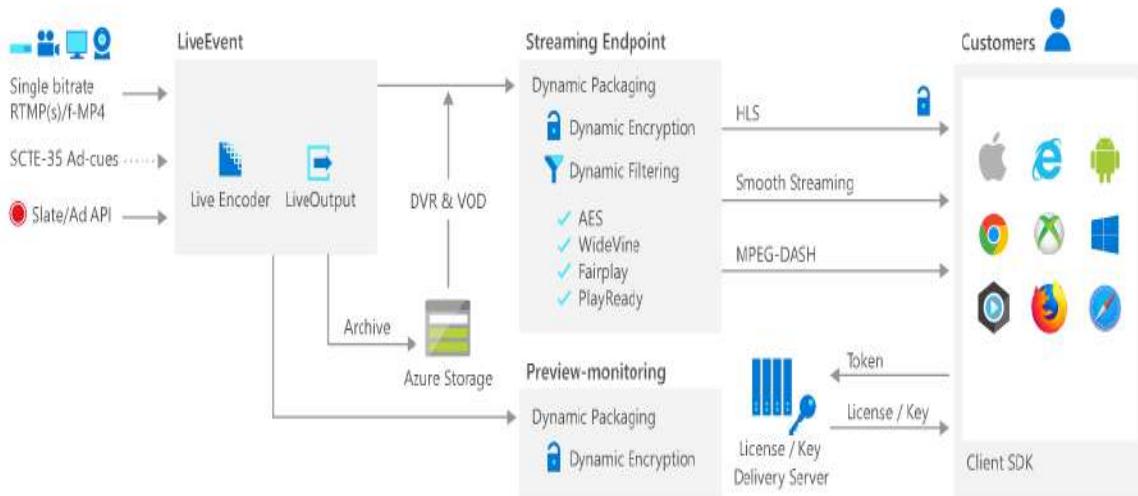
## Features of Azure Search

- Powerful queries
- Multi-language support
- Search suggestions
- Hit highlighting
- Faceted Navigation

Above are the different features associated with Azure search. In case if we want to have a cloud-based search engine that we can embed in our web application. Azure offers a service called Azure Search.

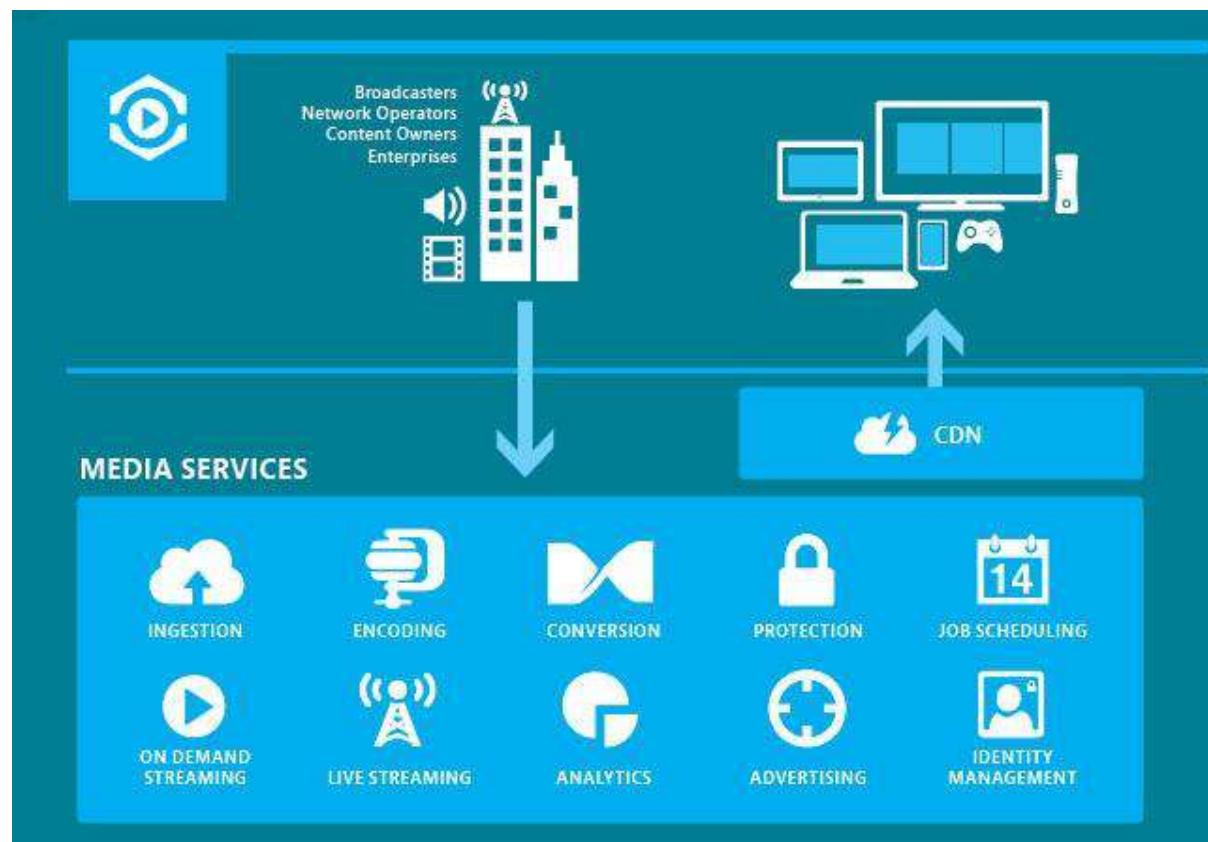
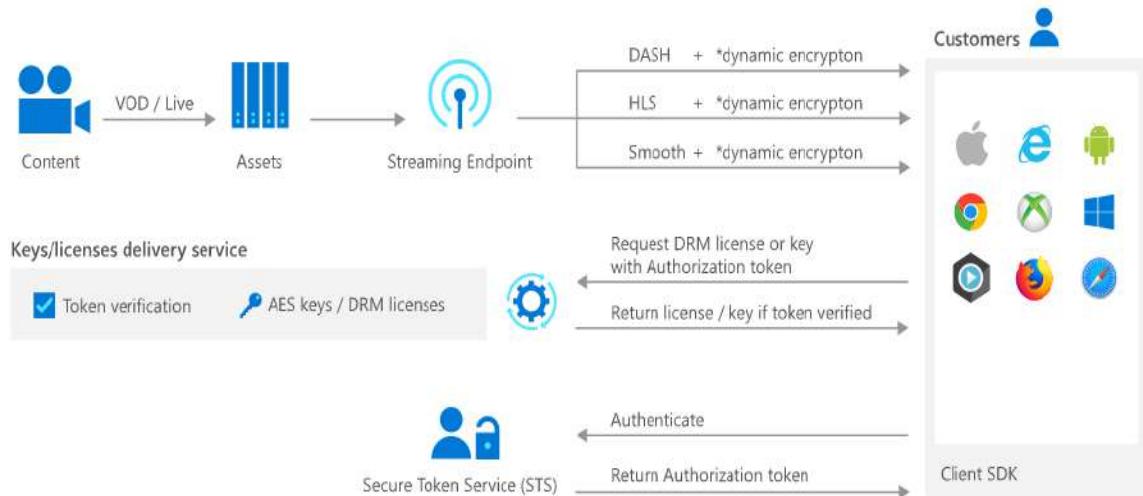
## Support your modern content distribution

Deliver a range of media source file and content streaming and protection formats to client technologies like HTTP Live Streaming (HLS), MPEG-DASH and Smooth Streaming. Use Azure Media Player to deliver content—applying industry standards like HTML5, media source extensions and encrypted media extensions and providing an enriched adaptive cloud streaming experience.



## Scale delivery according to your needs

Live broadcast a town hall or company meeting, a webinar, or a large sporting event to any online audience. Azure Media Services handles audiences of all sizes while you control the properties of the outgoing video livestream, such as how much is recorded and whether or not viewers can start watching.



Media services provides:

- 1) Ingestion
- 2) Encoding
- 3) Conversion
- 4) Protection
- 5) Job Scheduling
- 6) On Demand Streaming
- 7) Live Streaming
- 8) Analytics
- 9) Advertising
- 10) Identity Management

# AZURE MIGRATION

## Azure Migrate

Azure Migrate offers one stop solution to migrate non-Azure infrastructure to Azure. Using Azure Migrate, you can migrate Servers, Databases, Web Applications, Data and Virtual Desktops from almost any environment to Azure.

Azure migrate is not just a migration tool. You can perform discovery, assessment, dependency check, performance bench-marking of your infrastructure before you migrate your infrastructure to Azure.

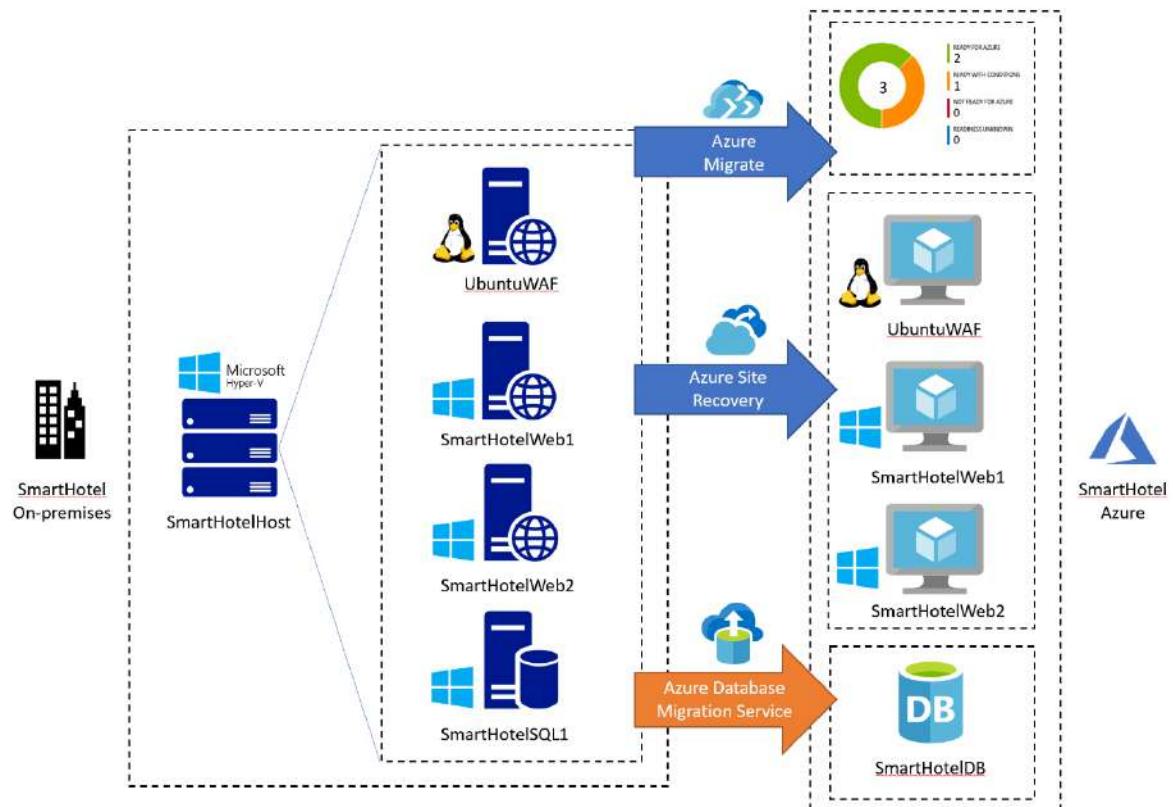


## What Is Azure Migrate?

Azure Migrate is an Azure migration program that helps organizations assess and automate migration of on-premises infrastructure, data and applications to the public cloud. This includes assessing the hosting costs and expected performance in the Azure cloud.

Azure Migrate offers the following features:

- **Centralized migration hub**—a unified platform that lets you plan, implement and track your migration strategy.
- **Varied toolkits**—including tools for assessing an existing server for migration (Server Assessment) and carrying out migration (Server Migration). Azure Migrate integrates with multiple Azure services and third-party tools.
- **Assessment and migration of workloads**—you can assess and migrate a variety of components, including servers, databases (i.e., Azure SQL Database), virtual desktop infrastructure (VDI) and moving web applications to Azure App Service.
- **Fast and cost-effective migration**—you can use Azure Data Box products to quickly and cost-effectively migrate a large amount of data to Azure.



What are the different migration patterns?

1. REHOST
2. REFACTOR
3. REARCHITECT
4. REBUILD

## 1. REHOST

Re-host simply means changing the host machine. This is the simplest method of migration as it does not need any redesigning of your architecture. It is also known as **lift-and-shift migration** – just move your application from on-premises to cloud as it is without any changes.

This type of pattern can be used when you need to move application quickly to the cloud without modifying it as it avoids downtime significantly.

## 2. REFACTOR

Refactoring is re-architecting the application, typically to use the benefits of cloud and cloud native features. It is simply re-packaging the application by changing the architecture.

This is a great method to implement when you are planning to use any new service provided by Microsoft Azure (For example, Azure DevOps). Refactoring can also be beneficial while migrating an existing application to **Azure Kubernetes Service or Azure Apps**.

## 3. REARCHITECT

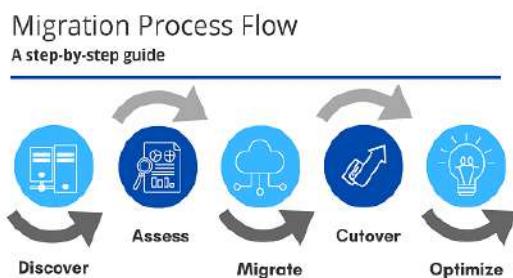
Rearchitecting is completely changing the architecture of the entire application. In re-architecting, we typically break down a monolithic application into different groups of microservices that work together and which can be scaled easily.

You can go for this migration pattern when you want to use existing application to meet the scalability requirements. New and innovative solutions can be implemented to minimise the use of services.

## 4. REBUILD

As the term goes, this involved a complete rebuild the entire application. Here, we are not referring to any existing system or changing any configuration. In short, the entire application is built from scratch using Azure Cloud Technology.

Here you could build **Greenfield** (building a system from scratch without any dependencies) using cloud Technologies like **Azure Functions, Azure Cosmos DB**.



# **7 Key Components of Azure Migrate**

Azure Migrate offers several tools that can help you plan, manage and implement migrations.

## **1. Azure Migrate: Discovery and Assessment**

This tool enables the discovery of servers running on-premises or in other clouds, including databases, virtual machines based on VMware or Hyper-V, and bare metal servers. You can use it to evaluate what is required to move these workloads to Azure.

## **2. Azure Migrate: Server Migration**

This tool allows you to automatically migrate all of the above workloads to the Azure cloud. In some cases, you will need to address compatibility issues by applying fixes to the existing servers.

## **3. Data Migration Assistant**

This is a stand-alone tool for assessing Microsoft SQL Server for cloud migration. You can migrate SQL Server to several destinations on Azure, including regular Azure VMs, the Azure SQL Database managed service, and Azure SQL Managed Instances. It allows you to identify potential issues that may prevent successful migration, highlights unsupported features, identifies new features you might want to adopt upon migration, and helps you plan an appropriate database migration path.

## **4. Azure Database Migration Service**

This fully managed service can help you automatically migrate on-premise SQL Server databases to the same destinations on Azure. It minimizes downtime by automating the entire migration process and enabling smooth rollback in case of a problem.

## **5. Movere**

This software-as-a-service (SaaS) platform provides business intelligence for servers and workloads, by mapping out and analyzing an entire IT environment. It provides information that allows organizations to maintain visibility and control over environments as they partake in migration projects.

## **6. Web App Migration Assistant**

This tool is designed to assess on-premises web applications or websites in preparation for migration to the Azure App Service. This includes .NET and PHP web applications.

## **7. Azure Data Box**

Azure Data Box products are physical appliances that help you migrate large amounts of data from offline storage to Azure. The Data Box series includes an SSD disk with 8 TB of storage, a larger appliance with 100 TB of storage, and a Heavy Data Box that can store up to 1 PB of data and ship it to the Azure cloud.

# **Azure Migrate: Important Points**

**Azure Migrate comes in two versions.** The previous version supported discovery and assessment, but not migration. Current version supports discovery, assessment and migration. You cannot create any new project in the previous version, and Microsoft recommends to use the current version for discovery, assessment and migration. **For Server Migration, Azure Migrate offers two major tools**

- 1) Server Assessment tool**
- 2) Server Migration tool.**

You can use Microsoft native tools or third party tools which are integrated with Azure Migrate. These third party tools are referred as Independent Software Vendor (ISV) tools.

- **Azure Migrate is a free service.** However, if you use ISV tools for assessment or migration, you might incur charges depending on the tool which you are using. If you use Microsoft native (first party) assessment and migration tools, there is no cost associated with it. If you use Dependency visualization during assessment, then data will be stored at Log Analytics Workspace. You will incur standard Log Analytics Workspace charges after 180 days.
- **There is no SLA associated with Azure Migrate,** as it is a free service.
- **Azure Migrate supports replication over Internet or ExpressRoute.** Replication traffic goes over port 443.

# AZURE DATA BOX



## WHAT IS AZURE DATA BOX?

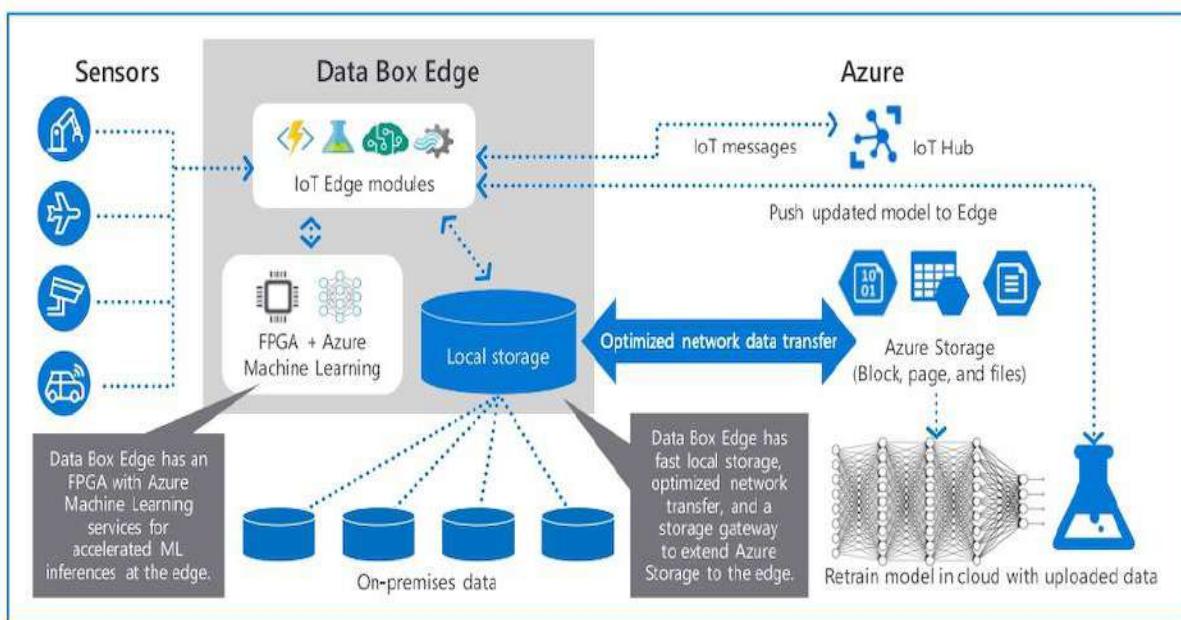
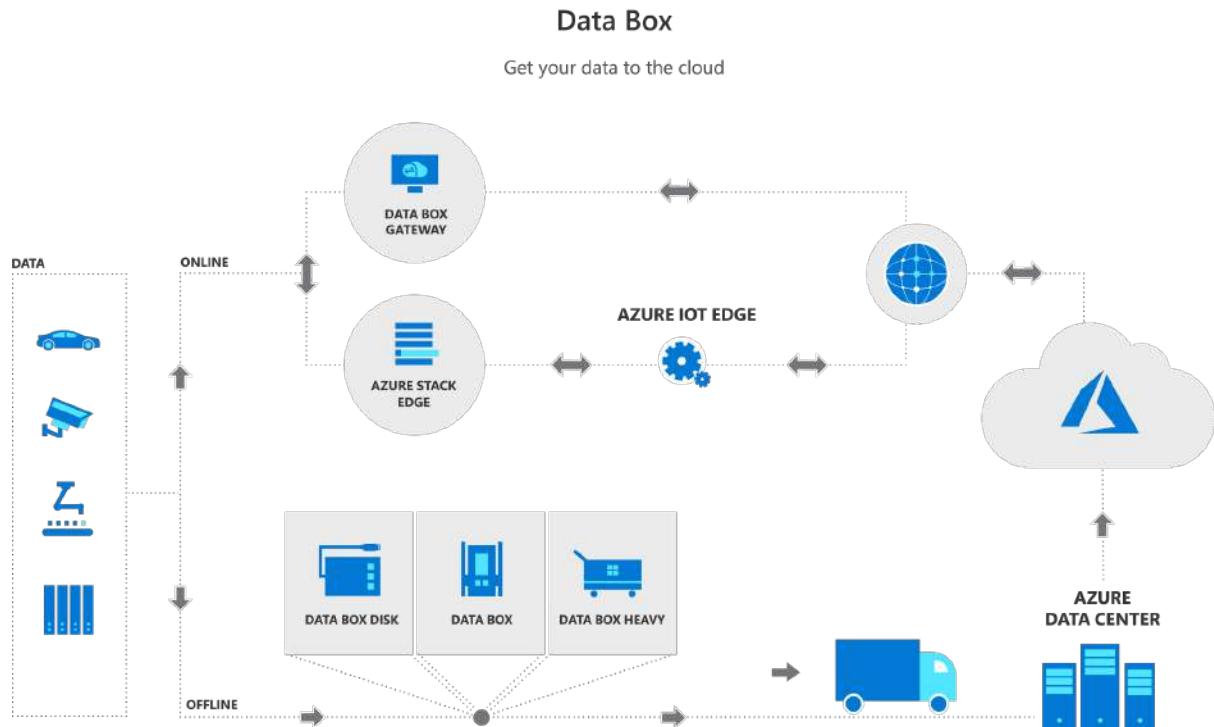
The Microsoft Azure Data Box cloud solution lets you send terabytes of data into and out of Azure in a quick, inexpensive, and reliable way. The secure data transfer is accelerated by shipping you a proprietary Data Box storage device.

The smallest option is called the Data Box and is in general availability status. It is a rugged device that allows organizations to have 100 TB of capacity on which to copy their data and then send it to be transferred to Azure. The device allows for 256-bit encryption on the data for safe transport, ensuring no data snooping or leaking during transport.

### Highlights include:

- 80 TB usable capacity per order
- One device per order
- Supports Azure Blob or Files
- Data can be copied to up to 10 storage accounts
- 1x1/10 Gbps RJ45, 2x10 Gbps SFP+ interface for accommodating various uplinks
- Data can be copied using NAS protocols such as SMB/NFS





## Use cases:

Data Box is ideally suited to transfer data sizes larger than 40 TBs in scenarios with no to limited network connectivity. The data movement can be one-time, periodic, or an initial bulk data transfer followed by periodic transfers.

Here are the various scenarios where Data Box can be used to import data to Azure.

- **One time migration** - when large amount of on-premises data is moved to Azure.
  - Moving a media library from offline tapes into Azure to create an online media library.
  - Migrating your VM farm, SQL server, and applications to Azure
  - Moving historical data to Azure for in-depth analysis and reporting using HDInsight
- **Initial bulk transfer** - when an initial bulk transfer is done using Data Box (seed) followed by incremental transfers over the network.
  - For example, backup solutions partners such as Commvault and Data Box are used to move initial large historical backup to Azure. Once complete, the incremental data is transferred via network to Azure storage.
- **Periodic uploads** - when large amount of data is generated periodically and needs to be moved to Azure. For example in energy exploration, where video content is generated on oil rigs and windmill farms.
- **Disaster recovery** - when a copy of the data from Azure is restored to an on-premises network. In a typical disaster recovery scenario, a large amount of Azure data is exported to a Data Box. Microsoft then ships this Data Box, and the data is restored on your premises in a short time.
- **Security requirements** - when you need to be able to export data out of Azure due to government or security requirements. For example, Azure Storage is available in US Secret and Top Secret clouds, and you can use Data Box to export data out of Azure.
- **Migrate back to on-premises or to another cloud service provider** - when you want to move all the data back to on-premises, or to another cloud service provider, export data via Data Box to migrate the workloads.

## The Workflow

A typical import flow includes the following steps:

1. **Order** - Create an order in the Azure portal, provide shipping information, and the destination Azure storage account for your data. If the device is available, Azure prepares and ships the device with a shipment tracking ID.
2. **Receive** - Once the device is delivered, cable the device for network and power using the specified cables. (The power cable is included with the device. You'll need to procure the data cables.) Turn on and connect to the device. Configure the device network and mount shares on the host computer from where you want to copy the data.
3. **Copy data** - Copy data to Data Box shares.

4. **Return** - Prepare, turn off, and ship the device back to the Azure datacenter.
5. **Upload** - Data is automatically copied from the device to Azure. The device disks are securely erased as per the National Institute of Standards and Technology (NIST) guidelines.

# AZURE DB MIGRATION SERVICE



## What is Azure DB Migration Service?

Azure Database Migration Service is a fully managed service designed to enable seamless migrations from multiple database sources to Azure Data platforms with minimal downtime. The service is currently in General Availability, with ongoing development efforts focused on:

- Reliability and performance.
- Iterative addition of source-target pairs.
- Continued investment in friction-free migrations.

How do I set up a Microsoft Azure Virtual Network?

While multiple Microsoft tutorials that can walk you through the process of setting up a virtual network, the official documentation appears in the article [Azure Virtual Network](#).

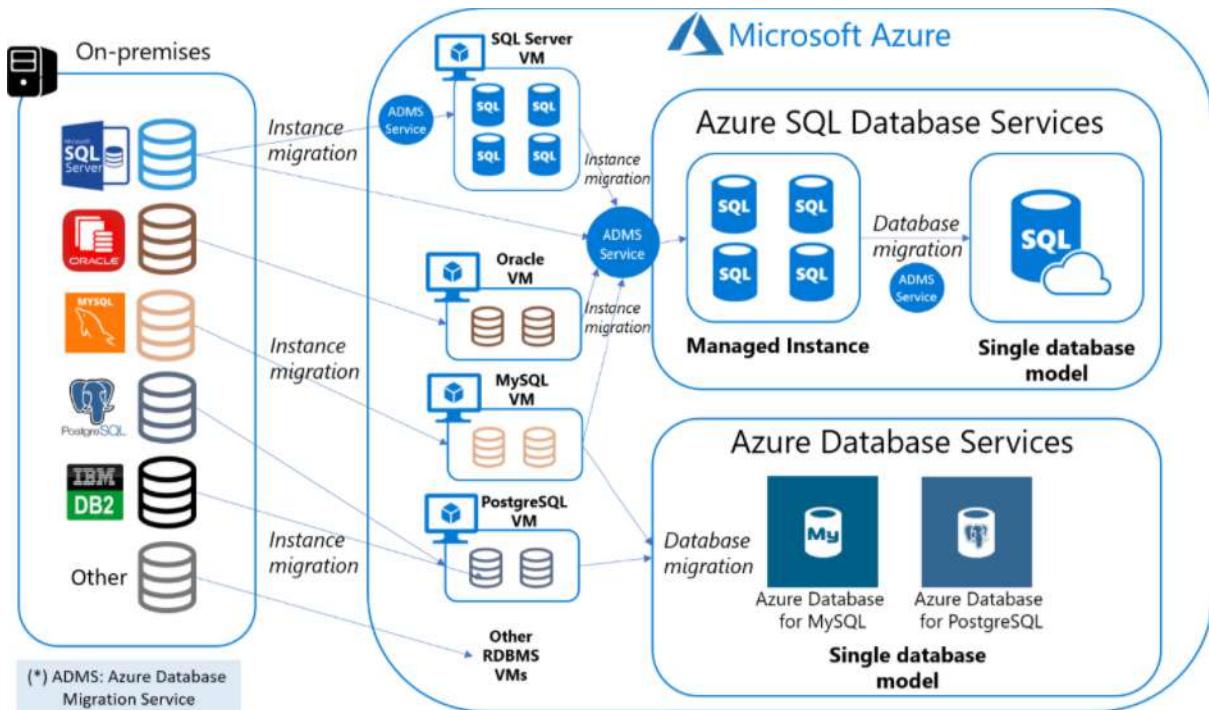
## Usage:

**What is a summary of the steps required to use Azure Database Migration Service to perform a database migration?**

During a typical, simple database migration, you:

1. Create a target database(s).
2. Assess your source database(s).
  - For homogenous migrations, assess your existing database(s) by using DMA.
  - For heterogeneous migrations (from compete sources), assess your existing database(s) with SSMA. You also use SSMA to convert database objects and migrate the schema to your target platform.
3. Create an instance of Azure Database Migration Service.

4. Create a migration project specifying the source database(s), target database(s), and the tables to migrate.
  5. Start the full load.
  6. Pick the subsequent validation.
7. Perform a manual switchover of your production environment to the new cloud-based database.

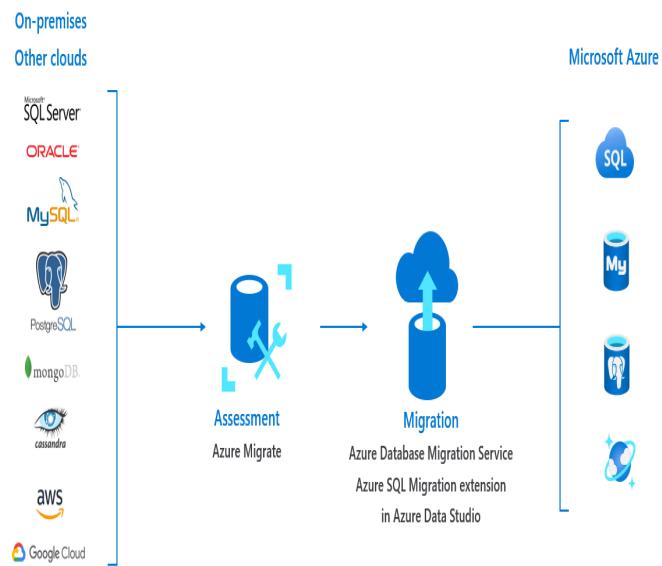


## USES

Move widely used databases

Migrate your data to Azure from the most common database management systems. Whether you're moving from an on-premises database or another cloud, Database Migration Service supports key migration scenarios such as SQL Server, MySQL, PostgreSQL, MongoDB, and Oracle.

# Tools and Services for your Migration Journey



## Automate your database migrations

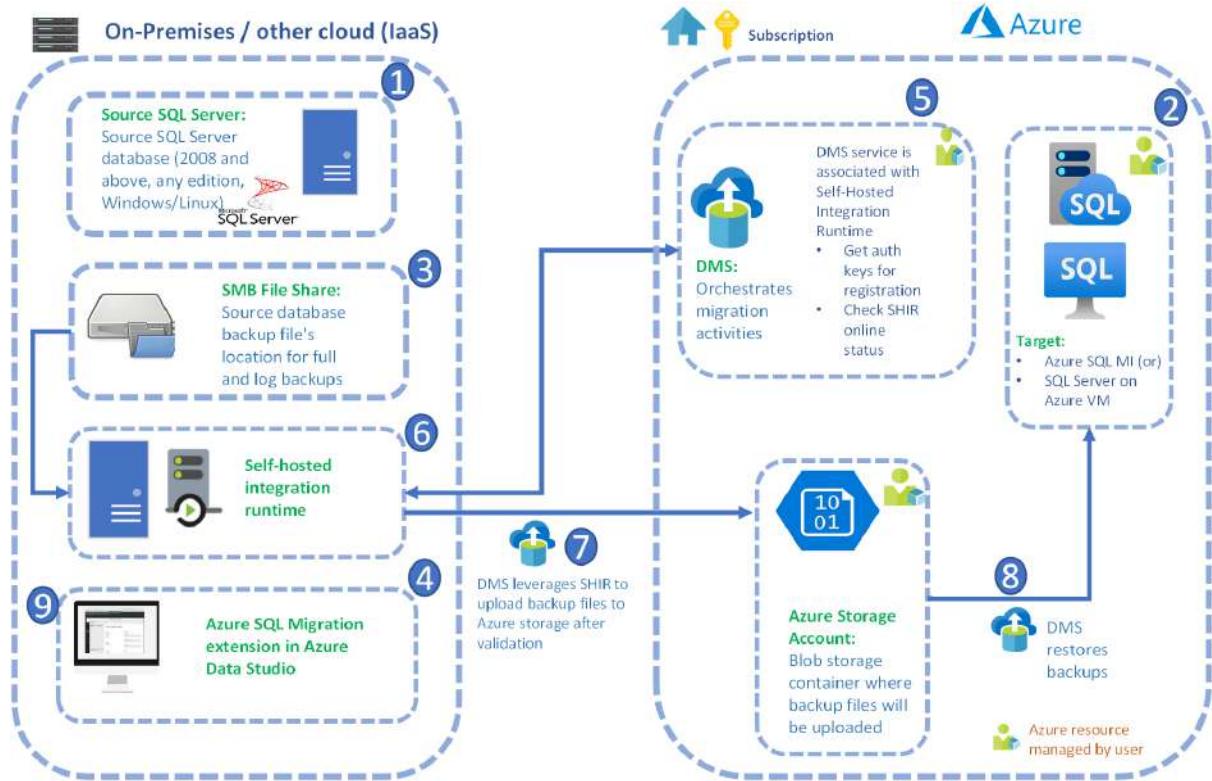
Save time and effort by automating your move to Azure with PowerShell. Database Migration Service works with PowerShell command lets to automatically migrate a list of databases.



## Architecture of Azure SQL Migration extension for Azure Data Studio:

Azure Database Migration Service (DMS) is one of the core components in the overall architecture. DMS provides a reliable migration orchestrator to enable database migrations to Azure SQL. Create or reuse an existing DMS using the Azure SQL Migration extension in Azure Data Studio(ADS). DMS uses Azure Data Factory's self-hosted integration runtime to access and upload valid backup files from your on-premises network share or your Azure Storage account.

The workflow of the migration process is illustrated below.



1. **Source SQL Server:** SQL Server instance on-premises, private cloud, or any public cloud virtual machine. All editions of SQL Server 2008 and above are supported.
2. **Target Azure SQL:** Supported Azure SQL targets are Azure SQL Managed Instance or SQL Server on Azure Virtual Machines (registered with SQL IaaS Agent extension in Full management mode)
3. **Network File Share:** Server Message Block (SMB) network file share where backup files are stored for the database(s) to be migrated. Azure Storage blob containers and Azure Storage file share are also supported.
4. **Azure Data Studio:** Download and install the Azure SQL Migration extension in Azure Data Studio.
5. **Azure DMS:** Azure service that orchestrates migration pipelines to do data movement activities from on-premises to Azure. DMS is associated with Azure

Data Factory's (ADF) self-hosted integration runtime (IR) and provides the capability to register and monitor the self-hosted IR.

6. **Self-hosted integration runtime (IR):** Self-hosted IR should be installed on a machine that can connect to the source SQL Server and the backup files location. DMS provides the authentication keys and registers the self-hosted IR.
7. **Backup files upload to Azure Storage:** DMS uses self-hosted IR to upload valid backup files from the on-premises backup location to your provisioned Azure Storage account. Data movement activities and pipelines are automatically created in the migration workflow to upload the backup files.
8. **Restore backups on target Azure SQL:** DMS restores backup files from your Azure Storage account to the supported target Azure SQL.

## WHAT IS AZURE SITE RECOVERY?



### WHAT IS AZURE SITE RECOVERY?

Azure site recovery is an excellent service from Microsoft that helps you to keep the workloads and business apps running in case of any outages or failures so that they won't affect the business. Basically, the service replicates the workloads running in your VMs to a different location from the primary site.

Now, in case of any failover in your primary site, it accesses the apps from the other location so that everything continues as it is. Once your primary site works as expected, you can failback to the prime site.

The overall aim is to safe and make available your workloads and the apps always even in case of any failures so that your business continues as it is.

## **Features of Azure Site Recovery**

There are many excellent features provided by Azure Site Recovery. Let's discuss here a few key features that azure site recovery provides.

### **Replication Of Azure VM**

**Azure Site Recovery** provides you the opportunity to configure disaster recovery for your Azure Virtual Machines from one location or region to another secondary location. So that in case of any disaster or failure, you can save your Data.

### **Replication for Workloads**

You can also able to replicate any of the workloads running on your Virtual machines or Physical servers so that it will not affect your business.

### **Quick BCDR Solution**

Using Azure Portal, you can quickly configure and manage the failover, replication and then you can fail back from a single place. If you are looking for a quick **BCDR Solution**, then Azure Site Recovery is one of the best options.

### **Customised Recovery Plans**

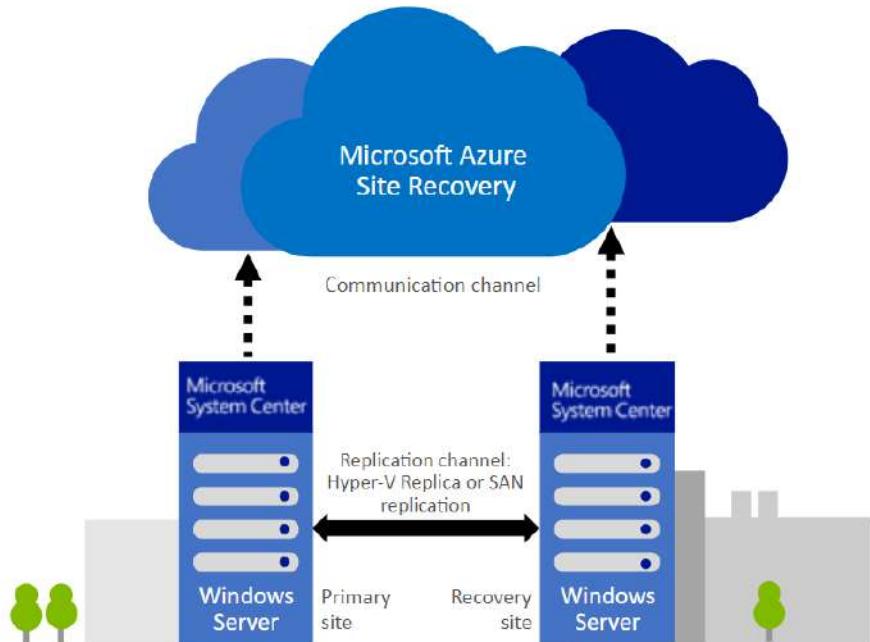
You can get the opportunity to customise the recovery plans based on your need like you can set the sequence of failover and recovery for multi-tier applications that are running on different virtual machines, etc. Not only that, but you can also easily integrate your recovery plans with your Azure Automation run books based on your need.

### **Seamless integration with BCDR technologies**

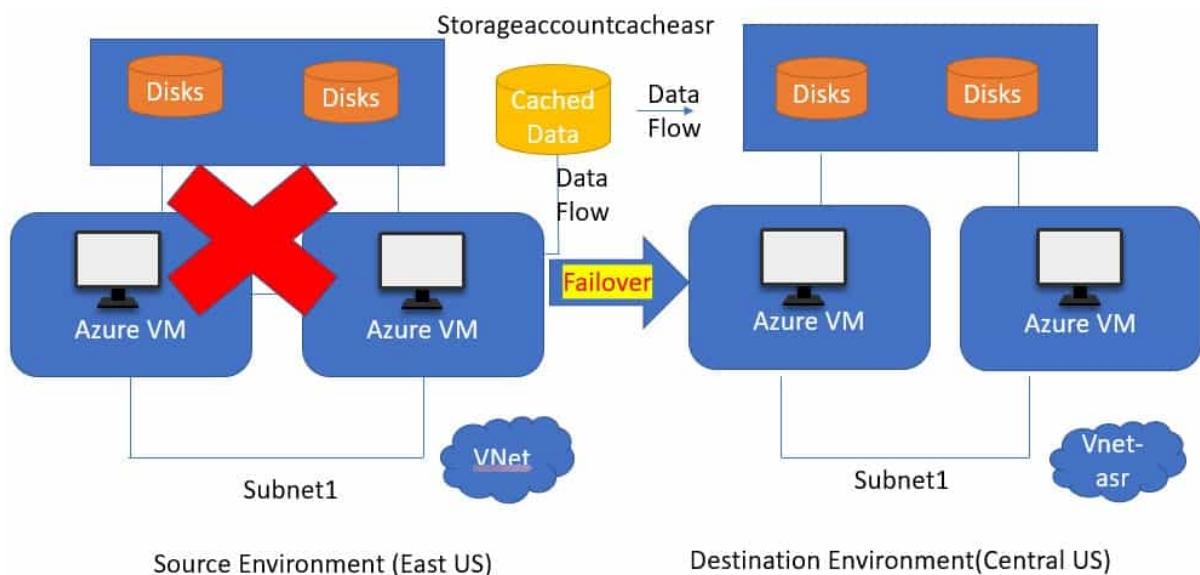
If you want to use any other **BCDR technologies**, Azure Site Recovery can seamlessly integrate with any other **BCDR technologies**.

## Replication Of Your On-premises VM

Azure Site Recovery can also help you to replicate your on-premises Virtual Machines and physical servers to Azure or to another data center that helps you to save a lot of costs.



## In Case of Failover



## **Main Components Of the Azure Site Recovery Architecture**

Below are the main components that are part of Azure Site Recovery architecture:

### **Source Environments**

The lists of main components for the source environments.

#### **Source Region Virtual Machines**

The list of Azure Virtual machines that are running on the supported source region with the supported operating systems.

#### **Storage for your Source Virtual Machines**

The managed or non managed disks of the Azure Virtual machines.

#### **VNets For your Source Virtual Machines**

The subnets inside the virtual network where your source Azure Virtual Machine is located.

#### **Cache storage account**

During the process of replication, the changes in the Virtual machines are initially stored in the Cache storage then the changes are being stored in the target storage. This is the reason Cache storage accounts play a vital role in the source Environment.

#### **Target subscription**

The subscription should be the same as the Source subscription.

#### **Resource Group For the Target**

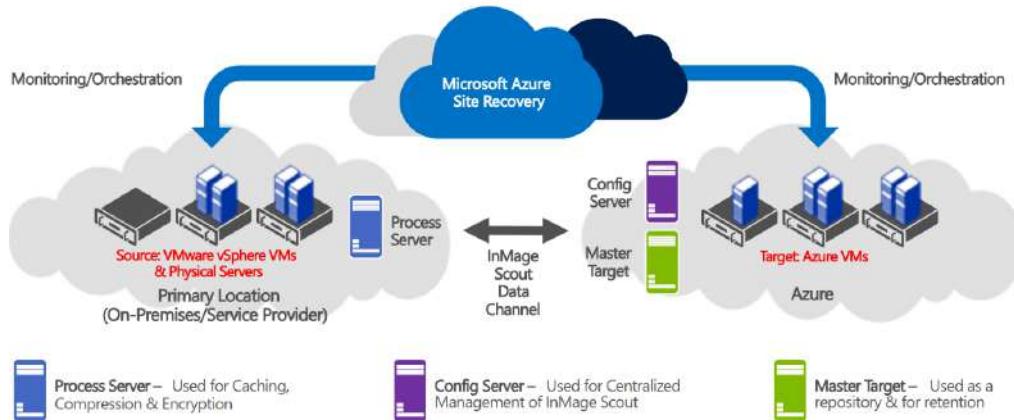
A new resource group is created by the Site Recovery in the destination region with the suffix “asr”.

#### **VNet For the Target**

The virtual network where your Virtual machines are going to store once after the failover.

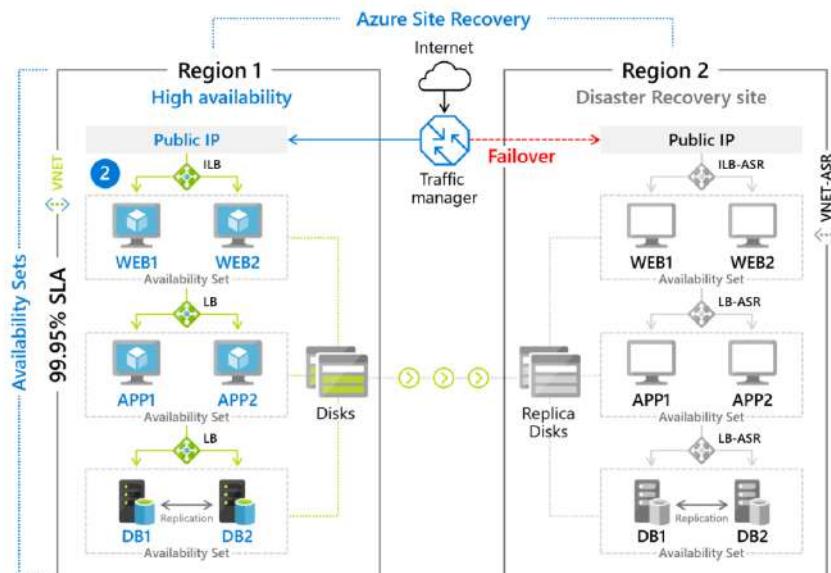
## Storage Account For the Target

The storage account where the data is going to replicate in the target environment in case your virtual machine is not using the managed disk.



## Simple to deploy and manage

Set up Azure Site Recovery simply by replicating an Azure VM to a different Azure region directly from the Azure portal. As a fully integrated offering, Site Recovery is automatically updated with new Azure features as they're released. Minimise recovery issues by sequencing the order of multi-tier applications running on multiple virtual machines. Ensure compliance by testing your disaster recovery plan without impacting production workloads or end users. And keep applications available during outages with automatic recovery from on-premises to Azure or Azure to another Azure region.



## Reduce infrastructure costs

Reduce the cost of deploying, monitoring, patching and maintaining on-premises disaster recovery infrastructure by eliminating the need for building or maintaining a costly secondary datacenter. Plus, you pay only for the compute resources you need to support your applications in Azure.



## Minimize downtime with dependable recovery

Easily comply with industry regulations such as ISO 27001 by enabling Site Recovery between separate Azure regions. Scale coverage to as many business-critical applications as you need, backed by Azure's service availability and support. Restore your most recent data quickly with Site Recovery.

# Microsoft Azure Site Recovery



**Reduce Cost • Less Complexity • Less Recovery Time**

# API MANAGEMENT

API Management (APIM) is a way to create consistent and modern API gateways for existing back-end services. API Management helps organizations publish APIs to external, partner, and internal developers to unlock the potential of their data and services. Businesses everywhere are looking to extend their operations as a digital platform, creating new channels, finding new customers and driving deeper engagement with existing ones. API Management provides the core competencies to ensure a successful API program through developer engagement, business insights, analytics, security, and protection. You can use Azure API Management to take any backend and launch a full-fledged API program based on it.

This provides an overview of common scenarios that involve APIM. It also gives a brief overview of the APIM system's main components. The article, then, gives a more detailed overview of each component.

To use API Management, administrators create APIs. Each API consists of one or more operations, and each API can be added to one or more products. To use an API, developers subscribe to a product that contains that API, and then they can call the API's operation, subject to any usage policies that may be in effect. Common scenarios include:

- **Securing mobile infrastructure** by gating access with API keys, preventing DOS attacks by using throttling, or using advanced security policies like JWT token validation.
- **Enabling ISV partner ecosystems** by offering fast partner onboarding through the developer portal and building an API facade to decouple from internal implementations that are not ripe for partner consumption.
- **Running an internal API program** by offering a centralized location for the organization to communicate about the availability and latest changes to APIs, gating access based on organizational accounts, all based on a secured channel between the API gateway and the backend.

The system is made up of the following components:

The **API gateway** is the endpoint that:

- Accepts API calls and routes them to your backends.
- Verifies API keys, JWT tokens, certificates, and other credentials.
- Enforces usage quotas and rate limits.
- Transforms your API on the fly without code modifications.
- Caches backend responses where set up.
- Logs call metadata for analytics purposes.

The **Azure portal** is the administrative interface where you set up your API program. Use it to:

- Define or import API schema.
- Package APIs into products.
- Set up policies like quotas or transformations on the APIs.
- Get insights from analytics.
- Manage users.

The **Developer portal** serves as the main web presence for developers, where they can:

- Read API documentation.
- Try out an API via the interactive console.
- Create an account and subscribe to get API keys.
- Access analytics on their own usage.

## APIs and Operations

APIs are the foundation of an API Management service instance. Each API represents a set of operations available to developers. Each API contains a reference to the back-end service that implements the API, and its operations map to the operations implemented by the back-end service. Operations in API Management are highly configurable, with control over URL mapping, query and path parameters, request and response content, and operation response caching. Rate limit, quotas, and IP restriction policies can also be implemented at the API or individual operation level.

## Products

Products are how APIs are surfaced to developers. Products in API Management have one or more APIs, and are configured with a title, description, and terms of use. Products can be **Open** or **Protected**. Protected products must be subscribed to before they can be used, while open products can be used without a subscription. When a product is ready for use by developers, it can be published. Once it is published, it can be viewed (and in the case of protected products subscribed to) by developers. Subscription approval is configured at the product level and can either require administrator approval, or be auto-approved.

Groups are used to manage the visibility of products to developers. Products grant visibility to groups, and developers can view and subscribe to the products that are visible to the groups in which they belong.

## Groups

Groups are used to manage the visibility of products to developers. API Management has the following immutable system groups:

- **Administrators** - Azure subscription administrators are members of this group. Administrators manage API Management service instances, creating the APIs, operations, and products that are used by developers.
- **Developers** - Authenticated developer portal users fall into this group. Developers are the customers that build applications using your APIs. Developers are granted access to the developer portal and build applications that call the operations of an API.
- **Guests** - Unauthenticated developer portal users, such as prospective customers visiting the developer portal of an API Management instance fall into this group. They can be granted certain read-only access, such as the ability to view APIs but not call them.

In addition to these system groups, administrators can create custom groups or leverage external groups in associated Azure Active Directory tenants. Custom and external groups can be used alongside system groups in giving developers visibility and access to API products. For example, you could create one custom group for developers affiliated with a specific partner organization and allow them access to the APIs from a product containing relevant APIs only. A user can be a member of more than one group.

For more information, see [How to create and use groups](#).

## Developers

Developers represent the user accounts in an API Management service instance. Developers can be created or invited to join by administrators, or they can sign up from the Developer portal. Each developer is a member of one or more groups, and can subscribe to the products that grant visibility to those groups.

When developers subscribe to a product, they are granted the primary and secondary key for the product. This key is used when making calls into the product's APIs.

For more information, see [How to create or invite developers](#) and [How to associate groups with developers](#).

## Policies

Policies are a powerful capability of API Management that allow the Azure portal to change the behavior of the API through configuration. Policies are a collection of statements that are executed sequentially on the request or response of an API. Popular statements include format conversion from XML to JSON and call rate limiting to restrict the number of incoming calls from a developer, and many other policies are available.

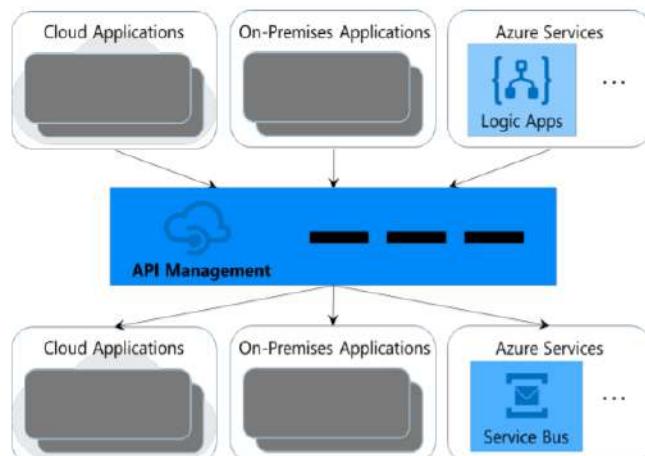
Policy expressions can be used as attribute values or text values in any of the API Management policies, unless the policy specifies otherwise. Some policies such as the Control flow and Set variable policies are based on policy expressions. For more information, see [Advanced policies and Policy expressions](#).

For a complete list of API Management policies, see [Policy reference](#). For more information on using and configuring policies, see [API Management policies](#). For a tutorial on creating a product with rate limit and quota policies, see [How to create and configure advanced product settings](#).

## Developer portal

The developer portal is where developers can learn about your APIs, view and call operations, and subscribe to products. Prospective customers can visit the developer portal, view APIs and operations, and sign up. The URL for your developer portal is located on the dashboard in the Azure portal for your API Management service instance.

You can customize the look and feel of your developer portal by adding custom content, customizing styles, and adding your branding.



**API Management exposes APIs from backend services to diverse clients**

## Event Grid

There are lots of integration scenarios in which communication through messages rather than API calls is the best approach. But requiring receiving software to periodically check whether a new message has arrived—commonly known as polling—can be wasteful. Why not let a receiver be notified via an event instead? This is exactly what Event Grid allows.

Rather than requiring a receiver to poll for new messages, the receiver instead registers an event handler for the event source it's interested in. Event Grid then invokes that event handler when the specified event occurs.

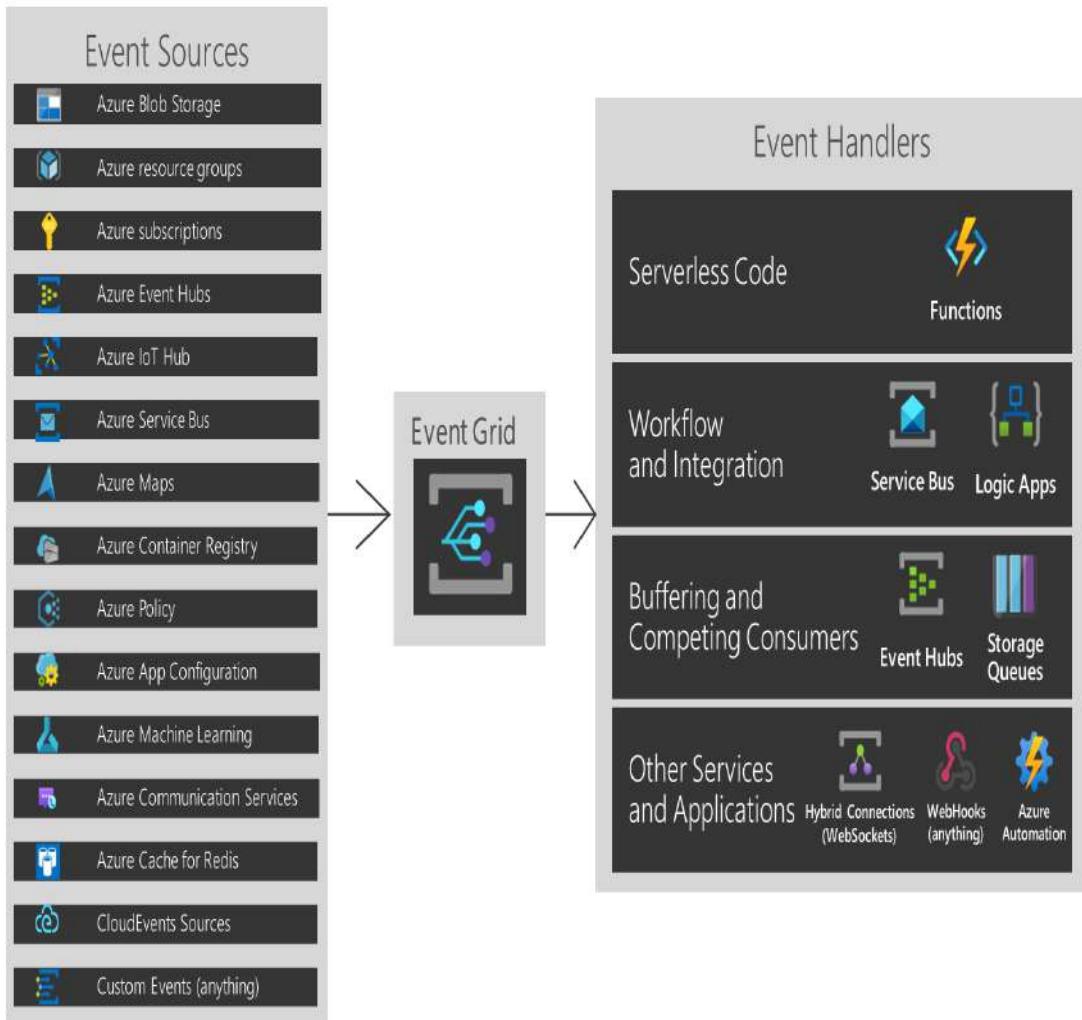
Azure Event Grid allows you to easily build applications with event-based architectures. First, select the Azure resource you would like to subscribe to, and then give the event handler or WebHook endpoint to send the event to.

Event Grid has built-in support for events coming from Azure services, like storage blobs and resource groups. Event Grid also has support for your own events, using custom topics.

You can use filters to route specific events to different endpoints, multicast to multiple endpoints, and make sure your events are reliably delivered.

Azure Event Grid is deployed to maximize availability by natively spreading across multiple fault domains in every region, and across availability zones (in regions that support them). For a list of regions that are supported by Event Grid, see Products available by region.

This article provides an overview of Azure Event Grid. If you want to get started with Event Grid, see Create and route custom events with Azure Event Grid.



## Event sources

Currently, the following Azure services support sending events to Event Grid. For more information about a source in the list, select the link.

- Azure App Configuration
- Azure Blob Storage
- Azure Communication Services
- Azure Container Registry
- Azure Event Hubs
- Azure IoT Hub
- Azure Key Vault
- Azure Machine Learning
- Azure Maps

- Azure Media Services
- Azure Policy
- Azure resource groups
- Azure Service Bus
- Azure SignalR
- Azure subscriptions
- Azure Cache for Redis
- Azure Kubernetes Service (preview)

## Event handlers

For full details on the capabilities of each handler as well as related articles, see event handlers. Currently, the following Azure services support handling events from Event Grid:

- Azure Automation
- Azure Functions
- Event Hubs
- Relay Hybrid Connections
- Logic Apps
- Power Automate (Formerly known as Microsoft Flow)
- Service Bus
- Queue Storage
- WebHooks

## Concepts

There are five concepts in Azure Event Grid that let you get going:

- **Events** - What happened.
- **Event sources** - Where the event took place.
- **Topics** - The endpoint where publishers send events.
- **Event subscriptions** - The endpoint or built-in mechanism to route events, sometimes to more than one handler. Subscriptions are also used by handlers to intelligently filter incoming events.
- **Event handlers** - The app or service reacting to the event.

For more information about these concepts, see Concepts in Azure Event Grid.

## Capabilities

Here are some of the key features of Azure Event Grid:

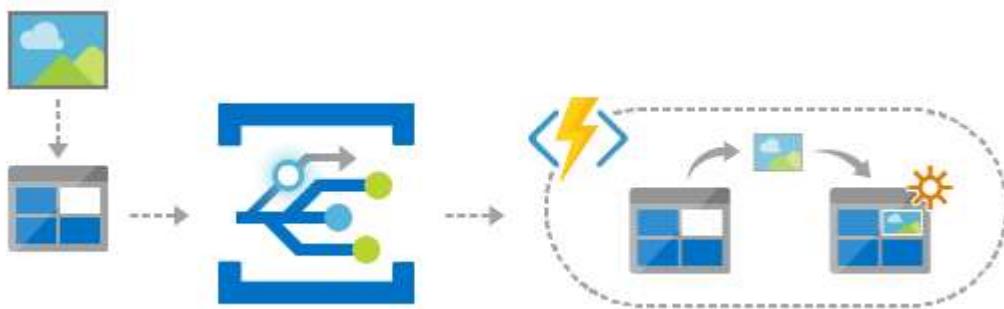
- **Simplicity** - Point and click to aim events from your Azure resource to any event handler or endpoint.
- **Advanced filtering** - Filter on event type or event publish path to make sure event handlers only receive relevant events.
- **Fan-out** - Subscribe several endpoints to the same event to send copies of the event to as many places as needed.
- **Reliability** - 24-hour retry with exponential backoff to make sure events are delivered.
- **Pay-per-event** - Pay only for the amount you use Event Grid.
- **High throughput** - Build high-volume workloads on Event Grid.
- **Built-in Events** - Get up and running quickly with resource-defined built-in events.
- **Custom Events** - Use Event Grid to route, filter, and reliably deliver custom events in your app.

For a comparison of Event Grid, Event Hubs, and Service Bus, see [Choose between Azure services that deliver messages](#).

## What can I do with Event Grid?

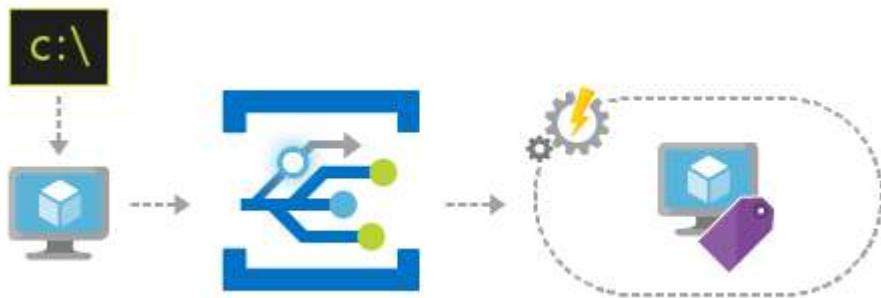
Azure Event Grid provides several features that vastly improve serverless, ops automation, and integration work:

### Serverless application architectures



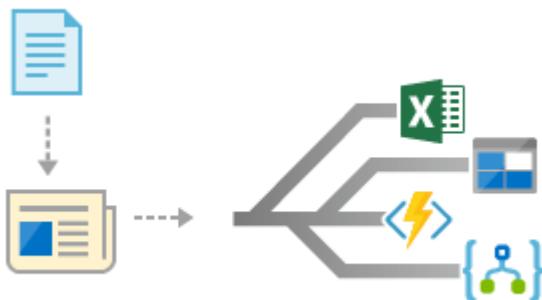
Event Grid connects data sources and event handlers. For example, use Event Grid to trigger a serverless function that analyzes images when added to a blob storage container.

## Ops Automation

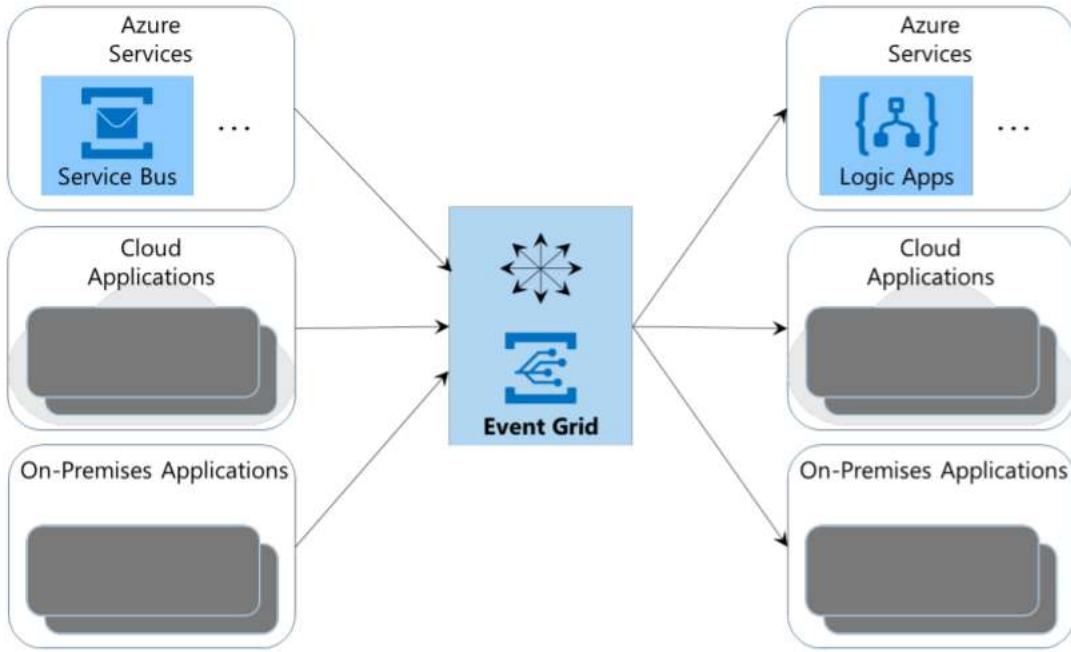


Event Grid allows you to speed automation and simplify policy enforcement. For example, use Event Grid to notify Azure Automation when a virtual machine or database in Azure SQL is created. Use the events to automatically check that service configurations are compliant, put metadata into operations tools, tag virtual machines, or file work items.

## Application integration



Event Grid connects your app with other services. For example, create a custom topic to send your app's event data to Event Grid, and take advantage of its reliable delivery, advanced routing, and direct integration with Azure. Or, you can use Event Grid with Logic Apps to process data anywhere, without writing code.



**Figure 9: Event Grid invokes receivers when a particular event has occurred.**

As the figure suggests, many Azure services can generate events. For example, the arrival of a new Service Bus message might cause Event Grid to send a message that starts a Logic App running, or the creation of a new blob in Azure Blob Storage might cause a custom cloud application to begin processing the contents of that blob. Using an event-driven approach can simplify application development, and it can also save money, since the receiver needn't waste cycles polling for new messages.

To receive an event from an Azure service, such as Blob Storage, the receiver subscribes to a standard topic provided for that service. But while subscribing to events from Azure services is probably today's most common use of Event Grid, it's not the only option. Cloud and on-premises applications can also create custom topics, letting Azure services and other software receive custom events by subscribing to these topics.

All of this raises an obvious question: Isn't Event Grid awfully similar to topics in Service Bus? Why does Azure Integration Services include both? The primary reason is that Service Bus is an enterprise messaging system, with all of the reliability and features that implies, while Event Grid provides only a simple and fast way to send events. Some of the differences that flow from these divergent goals are these:

- Service Bus topics work with messages, not events. They let receivers decide when to read a message, and they require the receiver to actively poll for new messages. With

Event Grid, events are raised by Azure Blob Storage or something else, then delivered to subscribers. No polling is required to receive these lightweight events.

- Event Grid is significantly more scalable than Service Bus, supporting up to 10,000,000 events per second in a single Azure region. To achieve this, Event Grid might deliver events out of order, while Service Bus guarantees in-order message delivery.
- Service Bus is fast, but Event Grid provides near real-time performance, with 99% of events delivered in less than a second.

Asynchronous communication is a necessary part of an iPaaS. By providing both Event Grid and Service Bus, Azure Integration Services addresses whatever communication needs your scenario might require.

## Health APIs

Azure Healthcare APIs is a set of managed API services based on open standards and frameworks that enable workflows to improve healthcare and offer scalable and secure healthcare solutions. Using a set of managed API services and frameworks that's dedicated to the healthcare industry is important and beneficial because health data collected from patients and healthcare consumers can be fragmented from across multiple systems, device types, and data formats. Gaining insights from health data is one of the biggest barriers to sustaining population and personal health and overall wellness understanding.

Bringing disparate systems, workflows, and health data together is more important today. A unified and aligned approach to health data access, standardization, and trend capturing would enable the discovery of operational and clinical insights. We can streamline the process of connecting new device applications and enable new research projects. Using Azure Healthcare APIs as a scalable and secure healthcare solution can enable workflows to improve healthcare through insights discovered by bringing protected health information (PHI) datasets together and connecting them end-to-end with tools for machine learning, analytics, and AI.

Azure Healthcare APIs provides the following benefits:

- Empower new workloads to leverage PHI by enabling the data to be collected and accessed in one place, in a consistent way.
- Discover new insight by bringing disparate PHI together and connecting it end-to-end with tools for machine learning, analytics, and AI.
- Build on a trusted cloud with confidence in how Protected Health Information is managed, stored, and made available. The new Microsoft Azure Healthcare APIs will, in addition to

FHIR, supports other healthcare industry data standards, like DICOM, extending healthcare data interoperability. The business model, and infrastructure platform has been redesigned to accommodate the expansion and introduction of different and future Healthcare data standards. Customers can use health data of different types across healthcare standards under the same compliance umbrella. Tools have been built into the managed service that allow customers to transform data from legacy or device proprietary formats, to FHIR. Some of these tools have been previously developed and open-sourced; Others will be net new.

#### **Azure Healthcare APIs enables you to:**

- Quickly connect disparate health data sources and formats such as structured, imaging, and device data and normalize it to be persisted in the cloud.
- Transform and ingest data into FHIR. For example, you can transform health data from legacy formats, such as HL7v2 or CDA, or from high frequency IoT data in device proprietary formats to FHIR.
- Connect your data stored in Healthcare APIs with services across the Azure ecosystem, like Synapse, and products across Microsoft, like Teams, to derive new insights through analytics and machine learning and to enable new workflows as well as connection to SMART on FHIR applications.
- Manage advanced workloads with enterprise features that offer reliability, scalability, and security to ensure that your data is protected, meets privacy and compliance certifications required for the healthcare industry.

#### **What are the key differences between Azure Healthcare APIs and Azure API for FHIR?**

#### **Linked Services**

The Azure Healthcare APIs now supports multiple health data standards for the exchange of structured data. A single collection of Azure Healthcare APIs enables you to deploy multiple instances of different service types (FHIR Service, DICOM Service, and IoT Connector) that seamlessly work with one another.

## Introducing DICOM Service

Azure Healthcare APIs now includes support for DICOM services. DICOM enables the secure exchange of image data and its associated metadata. DICOM is the international standard to transmit, store, retrieve, print, process, and display medical imaging information, and is the primary medical imaging standard accepted across healthcare. For more information about the DICOM Service, see [Overview of DICOM](#).

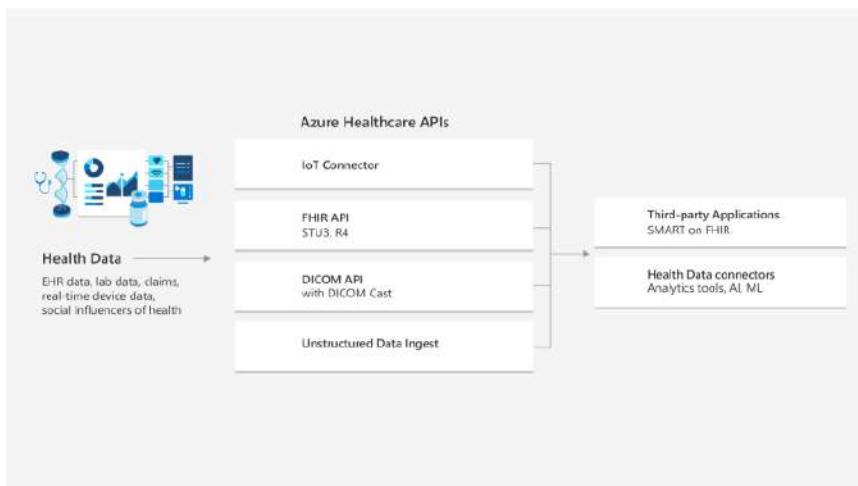
## Incremental changes to the FHIR Service

For the secure exchange of FHIR data, Healthcare APIs offers a few incremental capabilities that are not available in the Azure API for FHIR.

- Support for Transactions: In Healthcare APIs, the FHIR service supports transaction bundles. For more information about transaction bundles, visit [HL7.org](#) and refer to batch/transaction interactions.
- Chained Search Improvements: Chained Search & Reserve Chained Search are no longer limited by 100 items per sub query.

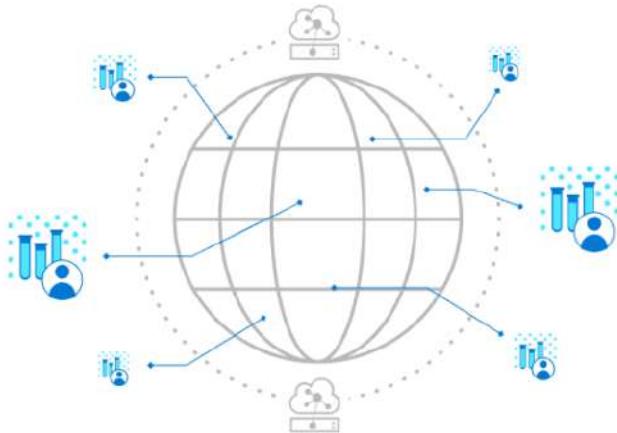
## Streamline health workloads

Unify data across patient outputs in the cloud to make PHI easier to exchange. Azure Healthcare APIs helps you standardize diverse data streams such as clinical, imaging, device, and unstructured data using our FHIR and DICOM services and Azure IoT Connector for FHIR. This produces better patient outcomes, improved clinical research scenarios, greater efficiency of clinical trials, and more informed care decisions.



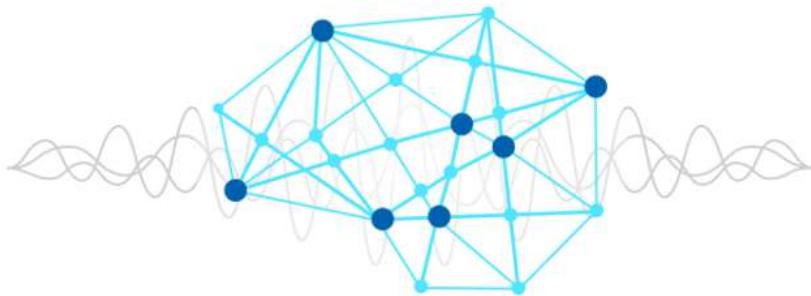
## Enable decentralized clinical trials and telehealth

Build a scalable end-to-end data pipeline that helps secure your PHI data workflows. Use Azure IoT Connector for FHIR to ingest biometric data from devices and standardize that data into FHIR to view in context with other clinical datasets. Use biometric data from devices to help drive decentralized clinical trials and telehealth.



## Derive insights from imaging

Manage imaging files in the cloud through APIs uniquely designed for DICOM. Easily store, query, retrieve, and exchange DICOM files and view clinical and imaging data together.



## Logic Apps

Azure Logic Apps is a cloud-based platform for creating and running automated workflows that integrate your apps, data, services, and systems. With this platform, you can quickly develop highly scalable integration solutions for your enterprise and business-to-business (B2B) scenarios. As a member of Azure Integration Services, Azure Logic Apps simplifies the way that you connect legacy, modern, and cutting-edge systems across cloud, on-premises, and hybrid environments.

The following list describes just a few example tasks, business processes, and workloads that you can automate using the Azure Logic Apps service:

- Schedule and send email notifications using Office 365 when a specific event happens, for example, a new file is uploaded.
- Route and process customer orders across on-premises systems and cloud services.
- Move uploaded files from an SFTP or FTP server to Azure Storage.
- Monitor tweets, analyze the sentiment, and create alerts or tasks for items that need review.

Based on the logic app resource type that you choose and create, your logic apps run in multi-tenant Azure Logic Apps, single-tenant Azure Logic Apps, or a dedicated integration service environment when accessing an Azure virtual network. To run logic apps in containers, create single-tenant based logic apps using Azure Arc enabled Logic Apps. For more information, review [What is Azure Arc enabled Logic Apps?](#) and [Resource type and host environment differences for logic apps](#).

To securely access and run operations in real time on various data sources, you can choose managed connectors from a 400+ and growing Azure connectors ecosystem to use in your workflows, for example:

- Azure services such as Blob Storage and Service Bus
- Office 365 services such as Outlook, Excel, and SharePoint
- Database servers such as SQL and Oracle
- Enterprise systems such as SAP and IBM MQ
- File shares such as FTP and SFTP

To communicate with any service endpoint, run your own code, organize your workflow, or manipulate data, you can use built-in triggers and actions, which run natively within the Azure Logic Apps service. For example, built-in triggers include Request, HTTP, and Recurrence. Built-in actions include Condition, For each, Execute JavaScript code, and operations that call Azure Functions, web apps or API apps hosted in Azure, and other Azure Logic Apps workflows.

For B2B integration scenarios, Azure Logic Apps includes capabilities from BizTalk Server. To define business-to-business (B2B) artifacts, you create integration account where you store these artifacts. After you link this account to your logic app, your workflows can use these B2B artifacts and exchange messages that comply with Electronic Data Interchange (EDI) and Enterprise Application Integration (EAI) standards.

For more information about the ways workflows can access and work with apps, data, services, and systems, review the following documentation:

- Connectors for Azure Logic Apps
- Managed connectors for Azure Logic Apps
- Built-in triggers and actions for Azure Logic Apps
- B2B enterprise integration solutions with Azure Logic Apps

## Key terms

The following terms are important concepts in the Azure Logic Apps service.

- **Logic app**

A logic app is the Azure resource you create when you want to develop a workflow. There are multiple logic app resource types that run in different environments.

- **Workflow**

A workflow is a series of steps that defines a task or process. Each workflow starts with a single trigger, after which you must add one or more actions.

- **Trigger**

A trigger is always the first step in any workflow and specifies the condition for running any further steps in that workflow. For example, a trigger event might be getting an email in your inbox or detecting a new file in a storage account.

- **Action**

An action is each step in a workflow after the trigger. Every action runs some operation in a workflow.

- **Built-in operations**

A built-in trigger or action is an operation that runs natively in Azure Logic Apps. For example, built-in operations provide ways for you to control your workflow's schedule or structure, run your own code, manage and manipulate data, send or receive requests to an endpoint, and complete other tasks in your workflow. Most built-in operations aren't associated with any service or system, but some built-in operations are available for specific services, such as Azure Functions or Azure App Service. Many also don't require that you first create a connection from your workflow and authenticate your identity. For more information and examples, review Built-in operations for Azure Logic Apps.

For example, you can start almost any workflow on a schedule when you use the Recurrence trigger. Or, you can have your workflow wait until called when you use the Request trigger.

## Managed connector

A managed connector is a prebuilt proxy or wrapper for a REST API that you can use to access a specific app, data, service, or system. Before you can use most managed connectors, you must first create a connection from your workflow and authenticate your identity. Managed connectors are published, hosted, and maintained by Microsoft. For more information, review Managed connectors for Azure Logic Apps.

For example, you can start your workflow with a trigger or run an action that works with a service such as Office 365, Salesforce, or file servers.

## Integration account

An integration account is the Azure resource you create when you want to define and store B2B artifacts for use in your workflows. After you create and link an integration account to your logic app, your workflows can use these B2B artifacts. Your workflows can also exchange messages that follow Electronic Data Interchange (EDI) and Enterprise Application Integration (EAI) standards.

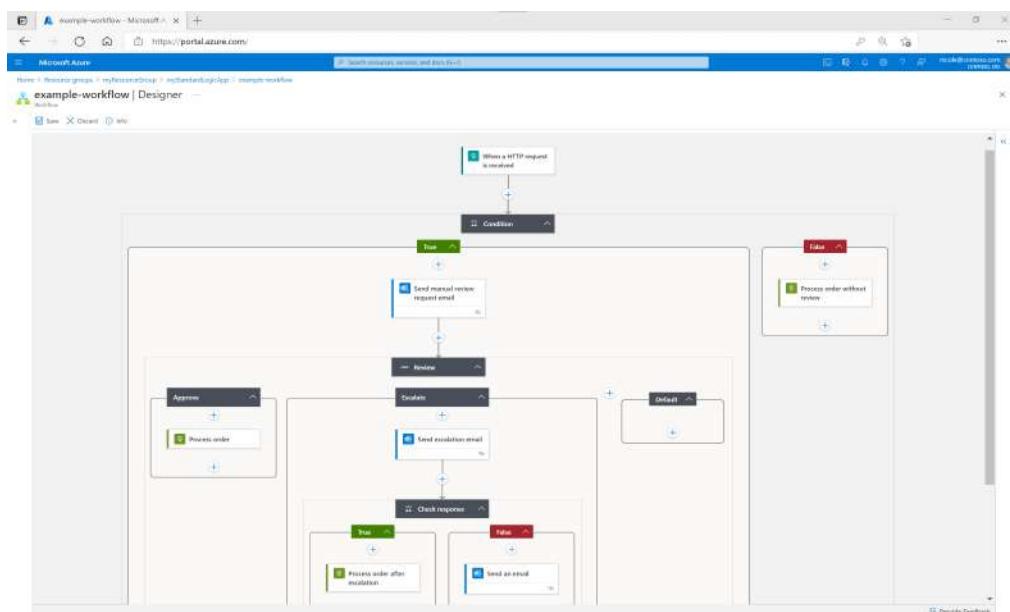
For example, you can define trading partners, agreements, schemas, maps, and other B2B artifacts. You can create workflows that use these artifacts and exchange messages over protocols such as AS2, EDIFACT, X12, and RosettaNet.

## How logic apps work

In a logic app, each workflow always starts with a single trigger. A trigger fires when a condition is met, for example, when a specific event happens or when data meets specific criteria. Many triggers include scheduling capabilities that control how often your workflow runs. Following the trigger, one or more actions run operations that, for example, process, handle, or convert data that travels through the workflow, or that advance the workflow to the next step.

The following screenshot shows part of an example enterprise workflow. This workflow uses conditions and switches to determine the next action. Let's say you have an order system, and your workflow processes incoming orders. You want to review orders above a certain cost manually. Your workflow already has previous steps that determine how much an incoming order costs. So, you create an initial condition based on that cost value. For example:

- If the order is below a certain amount, the condition is false. So, the workflow processes the order.
- If the condition is true, the workflow sends an email for manual review. A switch determines the next step.
- If the reviewer approves, the workflow continues to process the order.
- If the reviewer escalates, the workflow sends an escalation email to get more information about the order.
- If the escalation requirements are met, the response condition is true. So, the order is processed.
- If the response condition is false, an email is sent regarding the problem.



You can visually create workflows using the Azure Logic Apps workflow designer in the Azure portal, Visual Studio Code, or Visual Studio. Each workflow also has an underlying definition that's described using JavaScript Object Notation (JSON). If you prefer, you can edit workflows by changing this JSON definition. For some creation and management tasks, Azure Logic Apps provides Azure PowerShell and Azure CLI command support. For automated deployment, Azure Logic Apps supports Azure Resource Manager templates.

## Resource type and host environment differences

To create logic app workflows, you choose the Logic App resource type based on your scenario, solution requirements, the capabilities that you want, and the environment where you want to run your workflows.

The following table briefly summarizes differences between the original Logic App (Consumption) resource type and the Logic App (Standard) resource type. You'll also learn the differences between the single-tenant environment, multi-tenant environment, integration service environment (ISE), and App Service Environment v3 (ASEv3) for deploying, hosting, and running your logic app workflows.

RESOURCE TYPE AND HOST ENVIRONMENT DIFFERENCES				
Resource type	Benefits	Resource sharing and usage	Pricing and billing model	Limits management
Logic App (Consumption) Host environment : Multi-tenant Azure Logic Apps	- Easiest to get started - Pay-for-what-you-use - Fully managed	A single logic app can have <i>only one</i> workflow.  Logic apps created by customers <i>across multiple tenants</i> share the same processing (compute), storage, network, and so on.	Consumption (pay-per-execution)	Azure Logic Apps manages the default values for these limits, but you can change some of these values, if that option exists for a specific limit.

<b>RESOURCE TYPE AND HOST ENVIRONMENT DIFFERENCES</b>				
<b>Resource type</b>	<b>Benefits</b>	<b>Resource sharing and usage</b>	<b>Pricing and billing model</b>	<b>Limits management</b>
Logic App (Consumption)	- Enterprise scale for large workloads	A single logic app can have <i>only one</i> workflow.	ISE (fixed)	Azure Logic Apps manages the default values for these limits, but you can change some of these values, if that option exists for a specific limit.
Host environment : Integration service environment (ISE)	<ul style="list-style-type: none"> <li>- 20+ ISE-specific connectors that connect directly to virtual networks</li> <li>- Predictable pricing with included usage and customer-controlled scaling</li> <li>- Data stays in the same region where you deploy the ISE.</li> </ul>	Logic apps <i>in the same environment</i> share the same processing (compute), storage, network, and so on.		

RESOURCE TYPE AND HOST ENVIRONMENT DIFFERENCES				
Resource type	Benefits	Resource sharing and usage	Pricing and billing model	Limits management
<p>Logic App (Standard)</p> <p>Host environment : Single-tenant Azure Logic Apps</p> <p><b>Note:</b> If your scenario requires containers, create single-tenant based logic apps using Azure Arc enabled Logic Apps. For more information, review What is Azure Arc enabled Logic Apps?</p>	<ul style="list-style-type: none"> <li>- Run using the single-tenant Azure Logic Apps runtime.</li> <li>Deployment slots are currently not supported.</li> <li>- More built-in connectors for higher throughput and lower costs at scale</li> <li>- More control and fine-tuning capability around runtime and performance settings</li> <li>- Integrated support for virtual networks and private endpoints.</li> <li>- Create your own built-in connectors.</li> <li>- Data stays in the same region where you deploy your logic apps.</li> </ul>	<p>A single logic app can have multiple stateful and stateless workflows.</p> <p><i>Workflows in a single logic app and tenant</i> share the same processing (compute), storage, network, and so on.</p>	<p>Standard, based on a hosting plan with a selected pricing tier.</p> <p>If you run <i>stateful</i> workflows, which use external storage, the Azure Logic Apps runtime makes storage transactions that follow Azure Storage pricing.</p>	<p>You can change the default values for many limits, based on your scenario's needs.</p> <p><b>Important:</b> Some limits have hard upper maximums. In Visual Studio Code, the changes you make to the default limit values in your logic app project configuration files won't appear in the designer experience. For more information, see Edit app and environment settings for logic apps in single-tenant.</p>

RESOURCE TYPE AND HOST ENVIRONMENT DIFFERENCES				
Resource type	Benefits	Resource sharing and usage	Pricing and billing model	Limits management
Logic App (Standard) Host environment : App Service Environment v3 (ASEv3)	<p>Same capabilities as single-tenant <i>plus</i> the following benefits:</p> <ul style="list-style-type: none"> <li>- Fully isolate your logic apps.</li> <li>- Create and run more logic apps than in single-tenant Azure Logic Apps.</li> <li>- Pay only for the ASE App Service plan, no matter the number of logic apps that you create and run.</li> <li>- Can enable autoscaling or manually scale with more virtual machine instances or a different App Service plan.</li> <li>- Data stays in the same region where you deploy your logic apps.</li> <li>- Inherit the network setup from the selected ASEv3. For example, when deployed to an internal ASE, workflows can access the resources in a virtual network associated with the ASE and have internal IP addresses.</li> </ul>	<p>A single logic app can have multiple stateful and stateless workflows.</p> <p><i>Workflows in a single logic app and tenant</i> share the same processing (compute), storage, network, and so on.</p>	App Service plan	<p>You can change the default values for many limits, based on your scenario's needs.</p> <p><b>Important:</b> Some limits have hard upper maximums. In Visual Studio Code, the changes you make to the default limit values in your logic app project configuration files won't appear in the designer experience. For more information, see Edit app and environment settings for logic apps in single-tenant mode.</p>

## Why use Azure Logic Apps?

The Azure Logic Apps integration platform provides prebuilt Microsoft-managed API connectors and built-in operations so you can connect and integrate apps, data, services, and systems more easily and quickly. You can focus more on designing and implementing your solution's business logic and functionality, not on figuring out how to access your resources.

You usually won't have to write any code. However, if you do need to write code, you can create code snippets using Azure Functions and run that code from your workflow. You can also create code snippets that run in your workflow by using the **Inline Code** action. If your workflow needs to interact with events from Azure services, custom apps, or other solutions, you can monitor, route, and publish events using Azure Event Grid.

Azure Logic Apps is fully managed by Microsoft Azure, which frees you from worrying about hosting, scaling, managing, monitoring, and maintaining solutions built with these services. When you use these capabilities to create "serverless" apps and solutions, you can just focus on the business logic and functionality. These services automatically scale to meet your needs, make integrations faster, and help you build robust cloud apps using little to no code.

To learn how other companies improved their agility and increased focus on their core businesses when they combined Azure Logic Apps with other Azure services and Microsoft products, check out these customer stories.

The following sections provide more information about the capabilities and benefits in Azure Logic Apps.

## Visually create and edit workflows with easy-to-use tools

Save time and simplify complex processes by using the visual design tools in Azure Logic Apps. Create your workflows from start to finish by using the Azure Logic Apps workflow designer in the Azure portal, Visual Studio Code, or Visual Studio. Just start your workflow with a trigger, and add any number of actions from the connectors gallery.

If you're creating a multi-tenant based logic app, get started faster when you create a workflow from the templates gallery. These templates are available for common workflow patterns, which range from simple connectivity for Software-as-a-Service (SaaS) apps to advanced B2B solutions plus "just for fun" templates.

## **Connect different systems across various environments**

Some patterns and processes are easy to describe but hard to implement in code. The Azure Logic Apps platform helps you seamlessly connect disparate systems across cloud, on-premises, and hybrid environments. For example, you can connect a cloud marketing solution to an on-premises billing system, or centralize messaging across APIs and systems using Azure Service Bus. Azure Logic Apps provides a fast, reliable, and consistent way to deliver reusable and reconfigurable solutions for these scenarios.

## **Write once, reuse often**

Create your logic apps as Azure Resource Manager templates so that you can set up and automate deployments across multiple environments and regions.

## **First-class support for enterprise integration and B2B scenarios**

Businesses and organizations electronically communicate with each other by using industry-standard but different message protocols and formats, such as EDIFACT, AS2, X12, and RosettaNet. By using the enterprise integration capabilities supported by Azure Logic Apps, you can create workflows that transform message formats used by trading partners into formats that your organization's systems can interpret and process. Azure Logic Apps handles these exchanges smoothly and securely with encryption and digital signatures.

You can start small with your current systems and services, and then grow incrementally at your own pace. When you're ready, the Azure Logic Apps platform helps you implement and scale up to more mature integration scenarios by providing these capabilities and more:

- Integrate and build off Microsoft BizTalk Server, Azure Service Bus, Azure Functions, Azure API Management, and more.
- Exchange messages using EDIFACT, AS2, X12, and RosettaNet protocols.
- Process XML messages and flat files.
- Create an integration account to store and manage B2B artifacts, such as trading partners, agreements, transform maps, validation schemas, and more.

For example, if you use Microsoft BizTalk Server, your workflows can communicate with your BizTalk Server using the BizTalk Server connector. You can then run or extend BizTalk-like operations in your workflows by using integration account connectors. Going in the other direction, BizTalk Server can communicate with your workflows by using the Microsoft

BizTalk Server Adapter for Azure Logic Apps. Learn how to set up and use the BizTalk Server Adapter in your BizTalk Server.

## Built-in extensibility

If no suitable connector is available to run the code you want, you can create and call your own code snippets from your workflow by using Azure Functions. Or, create your own APIs and custom connectors that you can call from your workflows.

## Access resources inside Azure virtual networks

Logic app workflows can access secured resources, such as virtual machines (VMs) and other systems or services, that are inside an Azure virtual network when you create an integration service environment (ISE). An ISE is a dedicated instance of the Azure Logic Apps service that uses dedicated resources and runs separately from the global multi-tenant Azure Logic Apps service.

Running logic apps in your own dedicated instance helps reduce the impact that other Azure tenants might have on app performance, also known as the "noisy neighbors" effect. An ISE also provides these benefits:

- Your own static IP addresses, which are separate from the static IP addresses that are shared by the logic apps in the multi-tenant service. You can also set up a single public, static, and predictable outbound IP address to communicate with destination systems. That way, you don't have to set up extra firewall openings at those destination systems for each ISE.
- Increased limits on run duration, storage retention, throughput, HTTP request and response timeouts, message sizes, and custom connector requests. For more information, review Limits and configuration for Azure Logic Apps.
- When you create an ISE, Azure injects or deploys that ISE into your Azure virtual network. You can then use this ISE as the location for the logic apps and integration accounts that need access. For more information about creating an ISE, review Connect to Azure virtual networks from Azure Logic Apps.

## Pricing options

Each logic app type, which differs by capabilities and where they run (multi-tenant, single-tenant, integration service environment), has a different pricing model. For example, multi-tenant based logic apps use consumption pricing, while logic apps in an integration

service environment use fixed pricing. Learn more about pricing and metering for Azure Logic Apps.

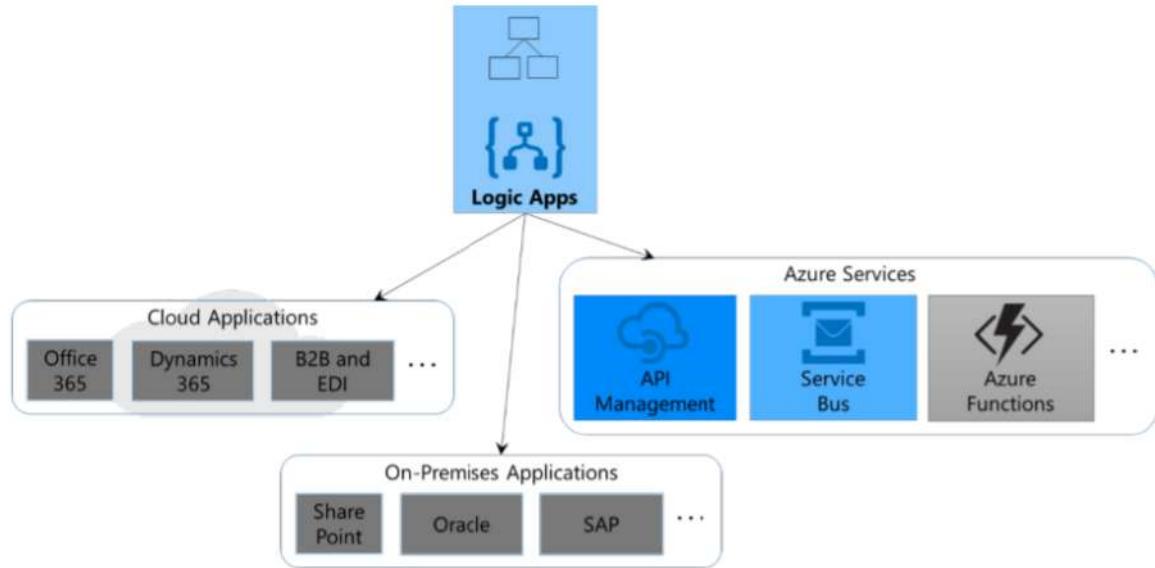


Figure 5: Logic Apps can access many kinds of software running in many different places.

## How does Azure Logic Apps differ from Functions, WebJobs, and Power Automate?

All these services help you connect and bring together disparate systems. Each service has their advantages and benefits, so combining their capabilities is the best way to quickly build a scalable, full-featured integration system. For more information, review Choose between Logic Apps, Functions, WebJobs, and Power Automate.

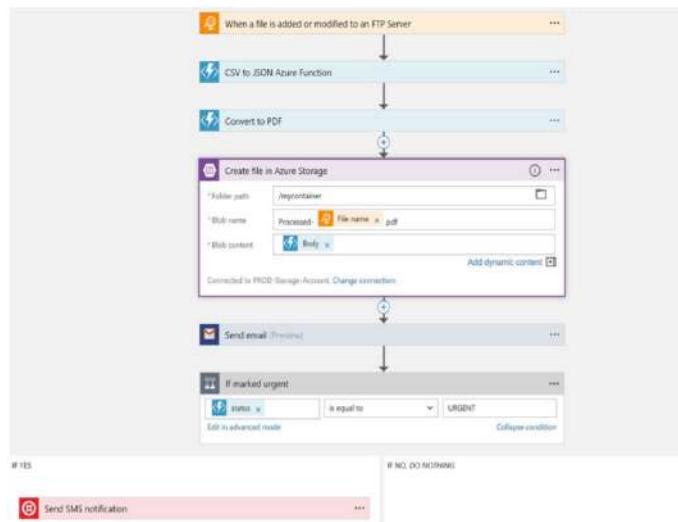


Figure 6: The Logic App Designer lets developers create logic apps without writing code.

As this example shows, the Logic App Designer lets its users create workflows by arranging actions on a surface. Like all logic apps, this one starts with a trigger, which in this case indicates that the workflow should run whenever a file is added to or changed on an FTP server. It then invokes two Azure functions, serverless chunks of code that transform the data in this file: first to JSON, then to PDF. In this example, the logic app's creator is in the process of defining the next action, which creates a file in Azure Blob Storage. After this, the workflow sends an email, then executes a conditional: If the newly added file is urgent, the logic app also sends a text.

This simple example doesn't access any cloud or on-premises applications—it mostly uses Azure services. Still, it illustrates the basics of how a developer creates a logic app. The key point is that writing code isn't required, letting developers implement business processes quickly and easily.

Logic Apps are at the heart of Azure's iPaaS offering, letting you easily connect the software and services you need to access. And because it allows you to integrate disparate services running both in the cloud and on-premises, Logic Apps provides an effective workflow solution for the world today.

## Notification Hubs

Azure Notification Hubs provide an easy-to-use and scaled-out push engine that enables you to send notifications to any platform (iOS, Android, Windows, etc.) from any back-end (cloud or on-premises). Notification Hubs works great for both enterprise and consumer scenarios. Here are a few example scenarios:

- Send breaking news notifications to millions with low latency.
- Send location-based coupons to interested user segments.
- Send event-related notifications to users or groups for media/sports/finance/gaming applications.
- Push promotional contents to applications to engage and market to customers.
- Notify users of enterprise events such as new messages and work items.
- Send codes for multi-factor authentication.

## **What are push notifications?**

Push notifications are a form of app-to-user communication where users of mobile apps are notified of certain desired information, usually in a pop-up or dialog box on a mobile device. Users generally choose to view or dismiss the message; choosing the former opens the mobile application that communicated the notification. Some notifications are silent - delivered behind the scenes for the app to process and decide what to do.

Push notifications are vital for consumer apps in increasing app engagement and usage, and for enterprise apps in communicating up-to-date business information. It's the best app-to-user communication because it is energy-efficient for mobile devices, flexible for the notifications senders, and available when corresponding applications are not active.

For more information on push notifications for a few popular platforms, see the following topics:

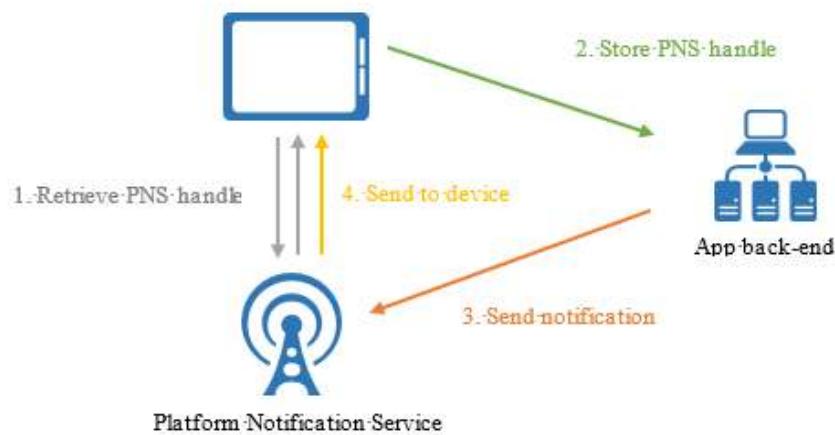
- Android
- iOS
- Windows

## **How do push notifications work?**

Push notifications are delivered through platform-specific infrastructures called *Platform Notification Systems* (PNSes). They offer basic push functionalities to deliver a message to a device with a provided handle, and have no common interface. To send a notification to all customers across the Android, iOS, and Windows versions of an app, the developer must work separately with Apple Push Notification Service (APNS), Firebase Cloud Messaging (FCM), and Windows Notification Service (WNS).

At a high level, here is how push works:

1. An application wants to receive a notification, so it contacts the PNS for the target platform on which the app is running and requests a unique and temporary push handle. The handle type depends on the system (for example, WNS uses URIs while APNS uses tokens).
2. The client app stores this handle in the app backend or provider.
3. To send a push notification, the app backend contacts the PNS using the handle to target a specific client app.
4. The PNS forwards the notification to the device specified by the handle.



## The Challenges of Push Notifications

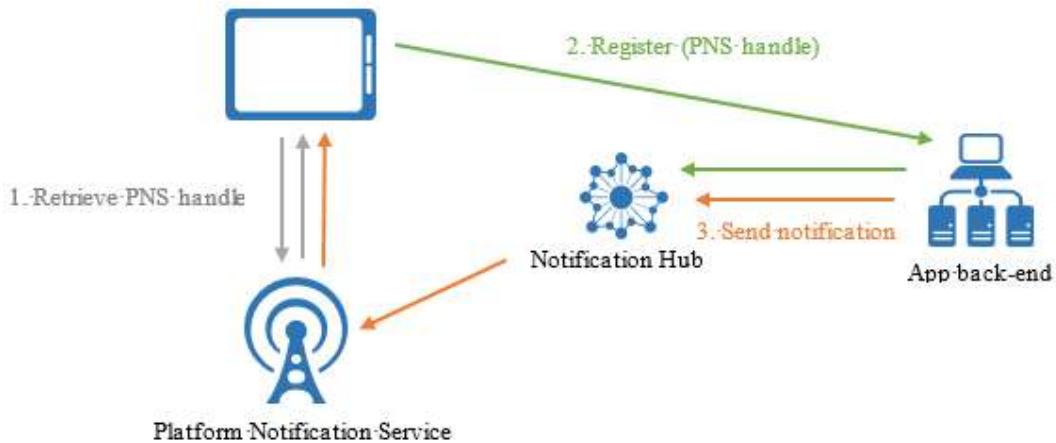
PNSes are powerful. However, they leave much work to the app developer to implement even common push notification scenarios, such as broadcasting push notifications to segmented users. Sending push notifications requires complex infrastructure that is unrelated to the application's main business logic. Some of the infrastructure challenges are:

- **Platform dependency**
  - The backend requires complex and hard-to-maintain platform-dependent logic to send notifications to devices on various platforms, as PNSes are not unified.
- **Scale**
  - Per PNS guidelines, device tokens must be refreshed on every app launch. The backend deals with a large amount of traffic and database access just to keep the tokens up-to-date. When the number of devices grows to hundreds, thousands, or millions, the cost of creating and maintaining this infrastructure is massive.
  - Most PNSes do not support broadcast to multiple devices. A simple broadcast to a million devices results in a million calls to the PNSes. Scaling this amount of traffic with minimal latency is nontrivial.
- **Routing**
  - Though PNSes provide a way to send messages to devices, most app notifications are targeted at users or interest groups. The backend must

maintain a registry to associate devices with interest groups, users, properties, etc. This overhead adds to the time to market and maintenance costs of an app.

## Why use Azure Notification Hubs?

Notification Hubs eliminates all complexities associated with sending push notifications on your own from your app backend. Its multi-platform, scaled-out push notification infrastructure reduces push-related coding and simplifies your backend. With Notification Hubs, devices are merely responsible for registering their PNS handles with a hub, while the backend sends messages to users or interest groups, as shown in the following figure:



Notification Hubs is your ready-to-use push engine with the following advantages:

- **Cross platforms**
  - Support for all major push platforms.
  - A common interface to push to all platforms in platform-specific or platform-independent formats with no platform-specific work.
- 1. Device handle management in one place.
- **Cross backends**
  - Cloud or on-premises.
  - .NET, Node.js, Java, Python, etc.

- **Rich set of delivery patterns**

- Broadcast to one or more platforms: You can instantly broadcast to millions of devices across platforms with a single API call.
- Push to device: You can target notifications to individual devices.
- Push to user: Tags and templates help you reach all cross-platform devices for a user.
- Push to segment with dynamic tags: The tags feature helps you segment devices and push to them according to your needs, whether you are sending to one segment or an expression of segments (For example, active AND lives in Seattle NOT new user). Instead of being restricted to publish-subscribe, you can update device tags anywhere and anytime.
- Localized push: The templates feature helps achieve localization without affecting backend code.
- Silent push: You can enable the push-to-pull pattern by sending silent notifications to devices and triggering them to complete certain pulls or actions.
- Scheduled push: You can schedule notifications to be sent anytime.
- Direct push: You can skip registering devices with the Notification Hubs service and directly batch push to a list of device handles.
- Personalized push: Device push variables help you send device-specific personalized push notifications with customized key-value pairs.

- **Rich telemetry**

- General push, device, error, and operation telemetry are available both in the Azure portal and programmatically.
- Per-message telemetry tracks each push from your initial request call to the Notification Hubs service successfully sending the pushes.

- Platform Notification System feedback communicates all feedback from PNSes to assist in debugging.

- **Scalability**

- Send fast messages to millions of devices without re-architecting or device sharding.

- **Security**

- Shared Access Secret (SAS) or federated authentication.

## Service Bus

Azure Service Bus is a fully managed enterprise message broker with message queues and publish-subscribe topics (in a namespace). Service Bus is used to decouple applications and services from each other, providing the following benefits:

- Load-balancing work across competing workers

- Safely routing and transferring data and control across service and application boundaries
- Coordinating transactional work that requires a high-degree of reliability.

Data is transferred between different applications and services using **messages**. A message is a container decorated with metadata, and contains data. The data can be any kind of information, including structured data encoded with the common formats such as the following ones: JSON, XML, Apache Avro, Plain Text.

Some common messaging scenarios are:

- **Messaging:** Transfer business data, such as sales or purchase orders, journals, or inventory movements.
- **Decouple applications:** Improve reliability and scalability of applications and services. Producer and consumer don't have to be online or readily available at the same time. The load is leveled such that traffic spikes don't overtax a service.
- **Load balancing:** Allow for multiple competing consumers to read from a queue at the same time, each safely obtaining exclusive ownership to specific messages.
- **Topics and subscriptions:** Enable 1:n relationships between publishers and subscribers, allowing subscribers to select particular messages from a published message stream.
- **Transactions:** Allows you to do several operations, all in the scope of an atomic transaction. For example, the following operations can be done in the scope of a transaction.
  - Obtain a message from one queue.
  - Post results of processing to one or more different queues.
  - Move the input message from the original queue.

The results become visible to downstream consumers only upon success, including the successful settlement of input message, allowing for once-only processing semantics. This transaction model is a robust foundation for the compensating transactions pattern in the greater solution context.

- **Message sessions:** Implement high-scale coordination of workflows and multiplexed transfers that require strict message ordering or message deferral.

If you're familiar with other message brokers like Apache ActiveMQ, Service Bus concepts are similar to what you know. As Service Bus is a platform-as-a-service (PaaS) offering, a key

difference is that you don't need to worry about the following actions. Azure takes care of those chores for you.

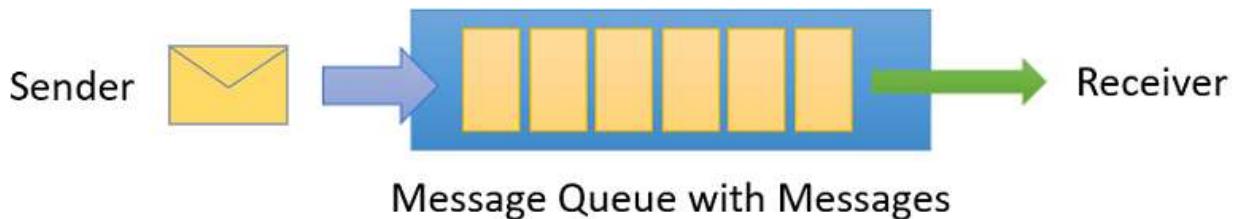
- Worrying about hardware failures
- Keeping the operating systems or the products patched
- Placing logs and managing disk space
- Handling backups
- Failing over to a reserve machine

## Concepts

This section discusses basic concepts of Service Bus.

## Queues

Messages are sent to and received from **queues**. Queues store messages until the receiving application is available to receive and process them.

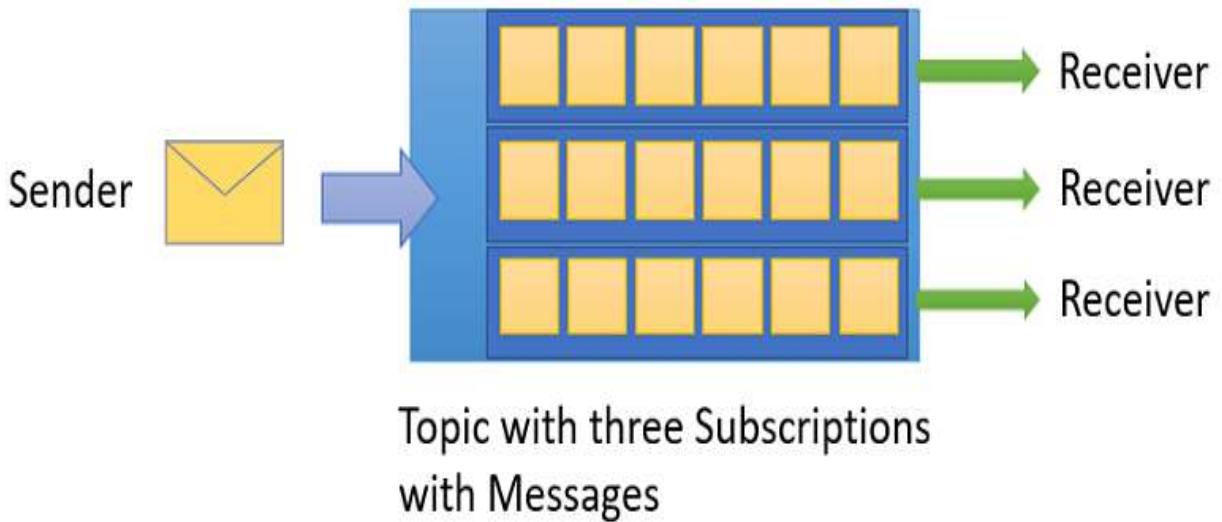


Messages in queues are ordered and timestamped on arrival. Once accepted by the broker, the message is always held durably in triple-redundant storage, spread across availability zones if the namespace is zone-enabled. Service Bus never leaves messages in memory or volatile storage after they've been reported to the client as accepted.

Messages are delivered in **pull** mode, only delivering messages when requested. Unlike the busy-polling model of some other cloud queues, the pull operation can be long-lived and only complete once a message is available.

## Topics

You can also use **topics** to send and receive messages. While a queue is often used for point-to-point communication, topics are useful in publish/subscribe scenarios.



Topics can have multiple, independent subscriptions, which attach to the topic and otherwise work exactly like queues from the receiver side. A subscriber to a topic can receive a copy of each message sent to that topic. Subscriptions are named entities. Subscriptions are durable by default, but can be configured to expire and then be automatically deleted. Via the Java Message Service (JMS) API, Service Bus Premium also allows you to create volatile subscriptions that exist for the duration of the connection.

You can define rules on a subscription. A subscription rule has a **filter** to define a condition for the message to be copied into the subscription and an optional **action** that can modify message metadata. For more information, see [Topic filters and actions](#). This feature is useful in the following scenarios:

- You don't want a subscription to receive all messages sent to a topic.
- You want to mark up messages with extra metadata when they pass through a subscription.

For more information about queues and topics, see [Service Bus queues, topics, and subscriptions](#).

## Namespaces

A namespace is a container for all messaging components (queues and topics). Multiple queues and topics can be in a single namespace, and namespaces often serve as application containers.

A namespace can be compared to a server in the terminology of other brokers, but the concepts aren't directly equivalent. A Service Bus namespace is your own capacity slice of a large cluster made up of dozens of all-active virtual machines. It may optionally span

three Azure availability zones. So, you get all the availability and robustness benefits of running the message broker at enormous scale. And, you don't need to worry about underlying complexities. Service Bus is serverless messaging.

## Advanced features

Service Bus also has advanced features that enable you to solve more complex messaging problems. The following sections describe these key features:

### Message sessions

To realize a first-in, first-out (FIFO) guarantee in Service Bus, use sessions. Message sessions enable joint and ordered handling of unbounded sequences of related messages.

### Auto-forwarding

The auto-forwarding feature enables you to chain a queue or subscription to another queue or topic that is part of the same namespace. When auto-forwarding is enabled, Service Bus automatically removes messages that are placed in the first queue or subscription (source) and puts them in the second queue or topic (destination).

### Dead-lettering

Service Bus supports a dead-letter queue (DLQ) to hold messages that cannot be delivered to any receiver, or messages that cannot be processed. You can then remove messages from the DLQ and inspect them.

### Scheduled delivery

You can submit messages to a queue or topic for delayed processing. For example, to schedule a job to become available for processing by a system at a certain time.

### Message deferral

When a queue or subscription client receives a message that it's willing to process, but for which processing isn't currently possible because of special circumstances within the

application, the entity can defer retrieval of the message to a later point. The message remains in the queue or subscription, but it's set aside.

## **Batching**

Client-side batching enables a queue or topic client to delay sending a message for a certain period of time. If the client sends more messages during this time period, it transmits the messages in a single batch.

## **Transactions**

A transaction groups two or more operations together into an execution scope. Service Bus supports grouping operations against a single messaging entity (queue, topic, subscription) within the scope of a transaction.

## **Filtering and actions**

Subscribers can define which messages they want to receive from a topic. These messages are specified in the form of one or more named subscription rules. For each matching rule condition, the subscription produces a copy of the message, which may be differently annotated for each matching rule.

## **Auto-delete on idle**

Auto-delete on idle enables you to specify an idle interval after which the queue is automatically deleted. The minimum duration is 5 minutes.

## **Duplicate detection**

If an error occurs that causes the client to have any doubt about the outcome of a send operation, duplicate detection takes the doubt out of these situations by enabling the sender to resend the same message, and the queue or topic discards any duplicate copies.

## **Shared access signature (SAS), Role-based access control, and managed identities**

Service Bus supports security protocols such as Shared Access Signatures (SAS), Role Based Access Control (RBAC) and Managed identities for Azure resources.

## **Geo-disaster recovery**

When Azure regions or datacenters experience downtime, Geo-disaster recovery enables data processing to continue operating in a different region or datacenter.

## **Security**

Service Bus supports standard Advanced Message Queuing Protocol (AMQP) 1.0 and HTTP/REST protocols.

## **Compliance with standards and protocols**

The primary wire protocol for Service Bus is Advanced Messaging Queueing Protocol (AMQP) 1.0, an open ISO/IEC standard. It allows customers to write applications that work against Service Bus and on-premises brokers such as ActiveMQ or RabbitMQ. The AMQP protocol guide provides detailed information in case you want to build such an abstraction.

Service Bus Premium is fully compliant with the Java/Jakarta EE Java Message Service (JMS) 2.0 API. And, Service Bus Standard supports the JMS 1.1 subset focused on queues. JMS is a common abstraction for message brokers and integrates with many applications and frameworks, including the popular Spring framework. To switch from other brokers to Azure Service Bus, you just need to recreate the topology of queues and topics, and change the client provider dependencies and configuration. For an example, see the ActiveMQ migration guide.

## **Client libraries**

Fully supported Service Bus client libraries are available via the Azure SDK.

- Azure Service Bus for .NET
- Azure Service Bus libraries for Java
- Azure Service Bus provider for Java JMS 2.0
- Azure Service Bus Modules for JavaScript and TypeScript
- Azure Service Bus libraries for Python

Azure Service Bus' primary protocol is AMQP 1.0 and it can be used from any AMQP 1.0 compliant protocol client. Several open-source AMQP clients have samples that explicitly demonstrate Service Bus interoperability. Review the AMQP 1.0 protocol guide to understand how to use Service Bus features with AMQP 1.0 clients directly.

## CLIENT LIBRARIES

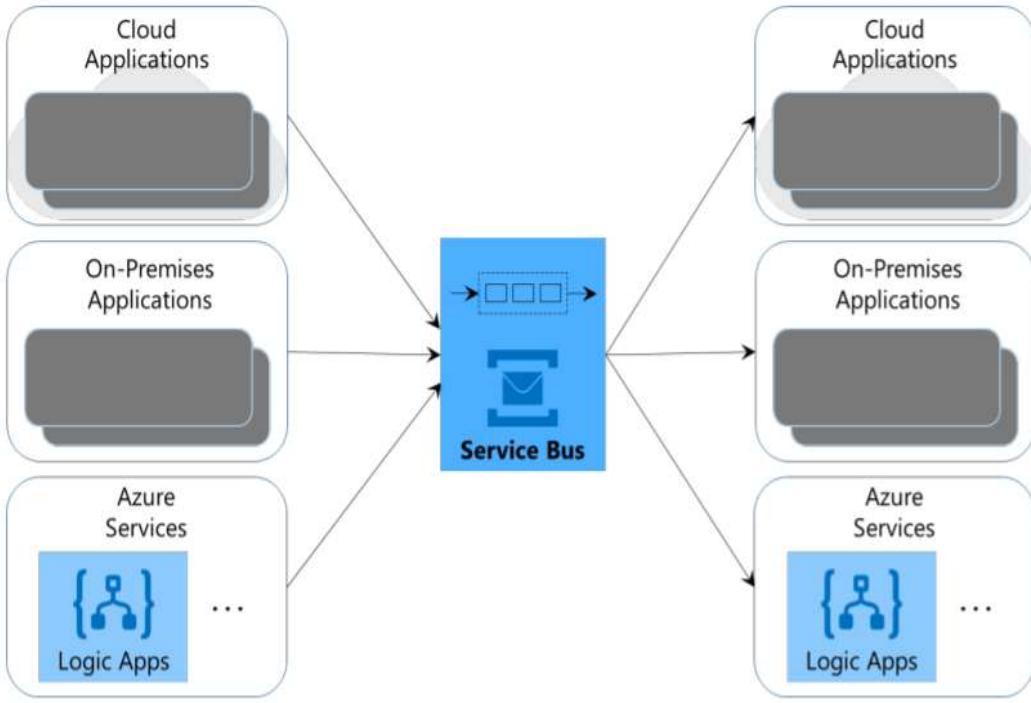
Language	Library
Java	Apache Qpid Proton-J
C/C++	Azure uAMQP C, Apache Qpid Proton-C
Python	Azure uAMQP for Python, Apache Qpid Proton Python
PHP	Azure uAMQP for PHP
Ruby	Apache Qpid Proton Ruby
Go	Azure Go AMQP, Apache Qpid Proton Go
C#/F#/VB	AMQP .NET Lite, Apache NMS AMQP
JavaScript/Node	Rhea

## Integration

Service Bus fully integrates with many Microsoft and Azure services, for instance:

- Event Grid
- Logic Apps
- Azure Functions
- Power Platform
- Dynamics 365
- Azure Stream Analytics

The essence of application integration is software talking to other software. But how should this communication happen? Sometimes, a direct call via API Management is perfect. In other cases, though, this synchronous style of communication won't work. What if both applications aren't available at the same time, for instance? For situations like this, an asynchronous approach is required. This kind of communication is exactly what Service Bus provides. Because it lets applications exchange messages through queues, Service Bus allows non-blocking interactions between different chunks of software.



**Service Bus Provides asynchronous communication between all kinds of software.**

## Web PubSub

The Azure Web PubSub Service helps you build real-time messaging web applications using WebSockets and the publish-subscribe pattern easily. This real-time functionality allows publishing content updates between server and connected clients (for example a single page web application or mobile application). The clients do not need to poll the latest updates, or submit new HTTP requests for updates.

### What is Azure Web PubSub service used for?

Any scenario that requires real-time publish-subscribe messaging between server and clients or among clients, can use Azure Web PubSub service. Traditional real-time features that often require polling from server or submitting HTTP requests, can also use Azure Web PubSub service.

Azure Web PubSub service can be used in any application type that requires real-time content updates. We list some examples that are good to use Azure Web PubSub service:

- **High frequency data updates:** gaming, voting, polling, auction.
- **Live dashboards and monitoring:** company dashboard, financial market data, instant sales update, multi-player game leader board, and IoT monitoring.
- **Cross-platform live chat:** live chat room, chat bot, on-line customer support, real-time shopping assistant, messenger, in-game chat, and so on.
- **Real-time location on map:** logistic tracking, delivery status tracking, transportation status updates, GPS apps.
- **Real-time targeted ads:** personalized real-time push ads and offers, interactive ads.
- **Collaborative apps:** coauthoring, whiteboard apps and team meeting software.
- **Push instant notifications:** social network, email, game, travel alert.
- **Real-time broadcasting:** live audio/video broadcasting, live captioning, translating, events/news broadcasting.
- **IoT and connected devices:** real-time IoT metrics, remote control, real-time status, and location tracking.
- **Automation:** real-time trigger from upstream events.

## **What are the benefits using Azure Web PubSub service?**

- **Built-in support for large-scale client connections and highly available architectures:**

Azure Web PubSub service is designed for large-scale real-time applications. The service allows multiple instances to work together and scale to millions of client connections. Meanwhile, it also supports multiple global regions for sharding, high availability, or disaster recovery purposes.

- **Support for a wide variety of client SDKs and programming languages:**

Azure Web PubSub service works with a broad range of clients, such as web and mobile browsers, desktop apps, mobile apps, server process, IoT devices, and game consoles. Since this service supports the standard WebSocket connection with publish-subscribe pattern, it is easily to use any standard WebSocket client SDK in different languages with this service.

- **Offer rich APIs for different messaging patterns:**

Azure Web PubSub service is a bi-directional messaging service that allows different messaging patterns among server and clients, for example:

- The server sends messages to a particular client, all clients, or a subset of clients that belong to a specific user, or have been placed in an arbitrary group.
- The client sends messages to clients that belong to an arbitrary group.
- The clients send messages to server.

## How to use the Azure Web PubSub service?

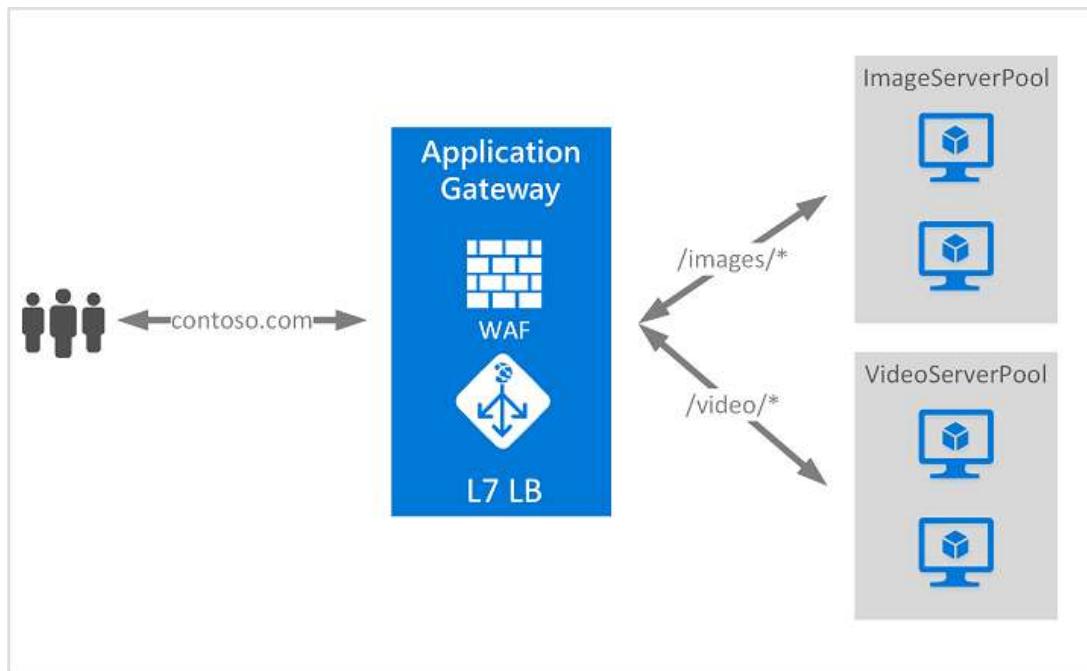
There are many different ways to program with Azure Web PubSub service, as some of the samples listed here:

- **Build serverless real-time applications:** Use Azure Functions' integration with Azure Web PubSub service to build serverless real-time applications in languages such as JavaScript, C#, Java and Python.
- **Use WebSocket subprotocol to do client-side only Pub/Sub** - Azure Web PubSub service provides WebSocket subprotocols to empower authorized clients to publish to other clients in a convenience manner.
- **Use provided SDKs to manage the WebSocket connections in self-host app servers** - Azure Web PubSub service provides SDKs in C#, JavaScript, Java and Python to manage the WebSocket connections easily, including broadcast messages to the connections, add connections to some groups, or close the connections, etc.
- **Send messages from server to clients via REST API** - Azure Web PubSub service provides REST API to enable applications to post messages to clients connected, in any REST capable programming languages.

# AZURE APPLICATION GATEWAY

Azure Application Gateway is a web traffic load balancer that enables you to manage traffic to your web applications. Traditional load balancers operate at the transport layer (OSI layer 4 - TCP and UDP) and route traffic based on source IP address and port, to a destination IP address and port.

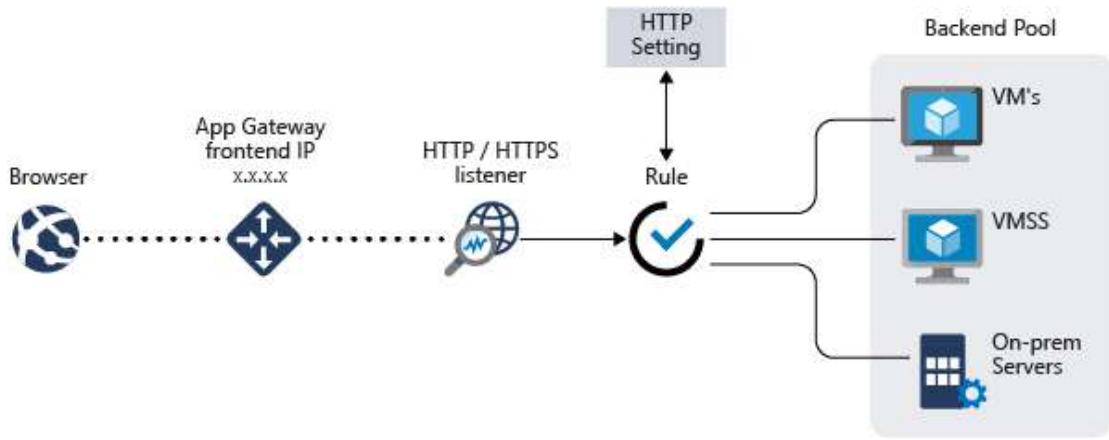
Application Gateway can make routing decisions based on additional attributes of an HTTP request, for example URI path or host headers. For example, you can route traffic based on the incoming URL. So if /images is in the incoming URL, you can route traffic to a specific set of servers (known as a pool) configured for images. If /video is in the URL, that traffic is routed to another pool that's optimized for videos.



This type of routing is known as application layer (OSI layer 7) load balancing. Azure Application Gateway can do URL-based routing and more.

## Azure Application Gateway Features

Azure Application Gateway features is a web traffic load balancer that enables you to manage traffic to your web applications.



Application Gateway includes the following features:

- Secure Sockets Layer (SSL/TLS) termination
- Autoscaling
- Zone redundancy
- Static VIP
- Web Application Firewall
- Ingress Controller for AKS
- URL-based routing
- Multiple-site hosting
- Redirection
- Session affinity
- Websocket and HTTP/2 traffic
- Connection draining
- Custom error pages
- Rewrite HTTP headers and URL
- Sizing

### **Secure Sockets Layer (SSL/TLS) termination**

Application gateway supports SSL/TLS termination at the gateway, after which traffic typically flows unencrypted to the backend servers. This feature allows web servers to be unburdened from costly encryption and decryption overhead.

But sometimes unencrypted communication to the servers isn't an acceptable option. This can be because of security requirements, compliance requirements, or the application may only accept a secure connection. For these applications, application gateway supports end to end SSL/TLS encryption.

## **Autoscaling**

Application Gateway Standard\_v2 supports autoscaling and can scale up or down based on changing traffic load patterns. Autoscaling also removes the requirement to choose a deployment size or instance count during provisioning.

## **Zone redundancy**

A Standard\_v2 Application Gateway can span multiple Availability Zones, offering better fault resiliency and removing the need to provision separate Application Gateways in each zone.

## **Static VIP**

The application gateway Standard\_v2 SKU supports static VIP type exclusively. This ensures that the VIP associated with application gateway doesn't change even over the lifetime of the Application Gateway.

## **Web Application Firewall**

Web Application Firewall (WAF) is a service that provides centralized protection of your web applications from common exploits and vulnerabilities. WAF is based on rules from the OWASP (Open Web Application Security Project) core rule sets 3.1 (WAF\_v2 only), 3.0, and 2.2.9.

Web applications are increasingly targets of malicious attacks that exploit common known vulnerabilities. Common among these exploits are SQL injection attacks, cross site scripting attacks to name a few. Preventing such attacks in application code can be challenging and may require rigorous maintenance, patching and monitoring at many layers of the application topology. A centralized web application firewall helps make security management much simpler and gives better assurance to application administrators against threats or intrusions. A WAF solution can also react to a security threat faster by patching a known vulnerability at a central location versus securing each of individual web applications. Existing application gateways can be converted to a Web Application Firewall enabled application gateway easily.

## Ingress Controller for AKS

Application Gateway Ingress Controller (AGIC) allows you to use Application Gateway as the ingress for an Azure Kubernetes Service (AKS) cluster.

The ingress controller runs as a pod within the AKS cluster and consumes Kubernetes Ingress Resources and converts them to an Application Gateway configuration, which allows the gateway to load-balance traffic to the Kubernetes pods. The ingress controller only supports Application Gateway Standard\_v2 and WAF\_v2 SKUs.

## URL-based routing

URL Path Based Routing allows you to route traffic to back-end server pools based on URL Paths of the request. One of the scenarios is to route requests for different content types to different pool.

For example, requests for `http://contoso.com/video/*` are routed to VideoServerPool, and `http://contoso.com/images/*` are routed to ImageServerPool. DefaultServerPool is selected if none of the path patterns match.

## Multiple-site hosting

With Application Gateway, you can configure routing based on host name or domain name for more than one web application on the same application gateway. It allows you to configure a more efficient topology for your deployments by adding up to 100+ websites to one application gateway. Each website can be directed to its own backend pool. For example, three domains, contoso.com, fabrikam.com, and adatum.com, point to the IP address of the application gateway. You'd create three multi-site listeners and configure each listener for the respective port and protocol setting.

Requests for `http://contoso.com` are routed to ContosoServerPool, `http://fabrikam.com` are routed to FabrikamServerPool, and so on.

Similarly, two subdomains of the same parent domain can be hosted on the same application gateway deployment. Examples of using subdomains could include `http://blog.contoso.com` and `http://app.contoso.com` hosted on a single application gateway deployment. For more information, see Application Gateway multiple site hosting.

You can also define wildcard host names in a multi-site listener and up to 5 host names per listener.

## Redirection

A common scenario for many web applications is to support automatic HTTP to HTTPS redirection to ensure all communication between an application and its users occurs over an encrypted path.

In the past, you may have used techniques such as dedicated pool creation whose sole purpose is to redirect requests it receives on HTTP to HTTPS. Application gateway supports the ability to redirect traffic on the Application Gateway. This simplifies application configuration, optimizes the resource usage, and supports new redirection scenarios, including global and path-based redirection. Application Gateway redirection support isn't limited to HTTP to HTTPS redirection alone. This is a generic redirection mechanism, so you can redirect from and to any port you define using rules. It also supports redirection to an external site as well.

Application Gateway redirection support offers the following capabilities:

- Global redirection from one port to another port on the Gateway. This enables HTTP to HTTPS redirection on a site.
- Path-based redirection. This type of redirection enables HTTP to HTTPS redirection only on a specific site area, for example a shopping cart area denoted by /cart/\*.
- Redirect to an external site.

## Session affinity

The cookie-based session affinity feature is useful when you want to keep a user session on the same server. By using gateway-managed cookies, the Application Gateway can direct subsequent traffic from a user session to the same server for processing. This is important in cases where session state is saved locally on the server for a user session.

## Websocket and HTTP/2 traffic

Application Gateway provides native support for the WebSocket and HTTP/2 protocols. There's no user-configurable setting to selectively enable or disable WebSocket support.

The WebSocket and HTTP/2 protocols enable full duplex communication between a server and a client over a long running TCP connection. This allows for a more interactive communication between the web server and the client, which can be bidirectional without the need for polling as required in HTTP-based implementations. These protocols have low overhead, unlike HTTP, and can reuse the same TCP connection for multiple request/responses resulting in a more efficient

resource utilization. These protocols are designed to work over traditional HTTP ports of 80 and 443.

## Connection draining

Connection draining helps you achieve graceful removal of backend pool members during planned service updates. This setting is enabled via the backend http setting and can be applied to all members of a backend pool during rule creation. Once enabled, Application Gateway ensures all deregistering instances of a backend pool don't receive any new request while allowing existing requests to complete within a configured time limit. This applies to both backend instances that are explicitly removed from the backend pool by a user configuration change, and backend instances that are reported as unhealthy as determined by the health probes. The only exception to this are requests bound for deregistering instances, which have been deregistered explicitly, because of gateway-managed session affinity and continues to be proxied to the deregistering instances.

## Custom error pages

Application Gateway allows you to create custom error pages instead of displaying default error pages. You can use your own branding and layout using a custom error page.

## Rewrite HTTP headers and URL

HTTP headers allow the client and server to pass additional information with the request or the response. Rewriting these HTTP headers helps you accomplish several important scenarios, such as:

- Adding security-related header fields like HSTS/ X-XSS-Protection.
- Removing response header fields that can reveal sensitive information.
- Stripping port information from X-Forwarded-For headers.

Application Gateway and WAF v2 SKU supports the capability to add, remove, or update HTTP request and response headers, while the request and response packets move between the client and back-end pools. You can also rewrite URLs, query string parameters and host name. With URL rewrite and URL path-based routing, you can choose to either route requests to one of the backend pools based on the original path or the rewritten path, using the re-evaluate path map option.

It also provides you with the capability to add conditions to ensure the specified headers or URL are rewritten only when certain conditions are met. These conditions are based on the request and response information.

## Sizing

Application Gateway Standard\_v2 can be configured for autoscaling or fixed size deployments. The v2 SKU doesn't offer different instance sizes. For more information on v2 performance and pricing, see [Autoscaling V2](#) and [Understanding pricing](#).

The Application Gateway Standard (v1) is offered in three sizes: Small, Medium, and Large. Small instance sizes are intended for development and testing scenarios. For a complete list of application gateway limits, see [Application Gateway service limits](#).

The following table shows an average performance throughput for each application gateway v1 instance with SSL offload enabled:

SIZING			
Average back-end page response size	Small	Medium	Large
6 KB	7.5 Mbps	13 Mbps	50 Mbps
100 KB	35 Mbps	100 Mbps	200 Mbps

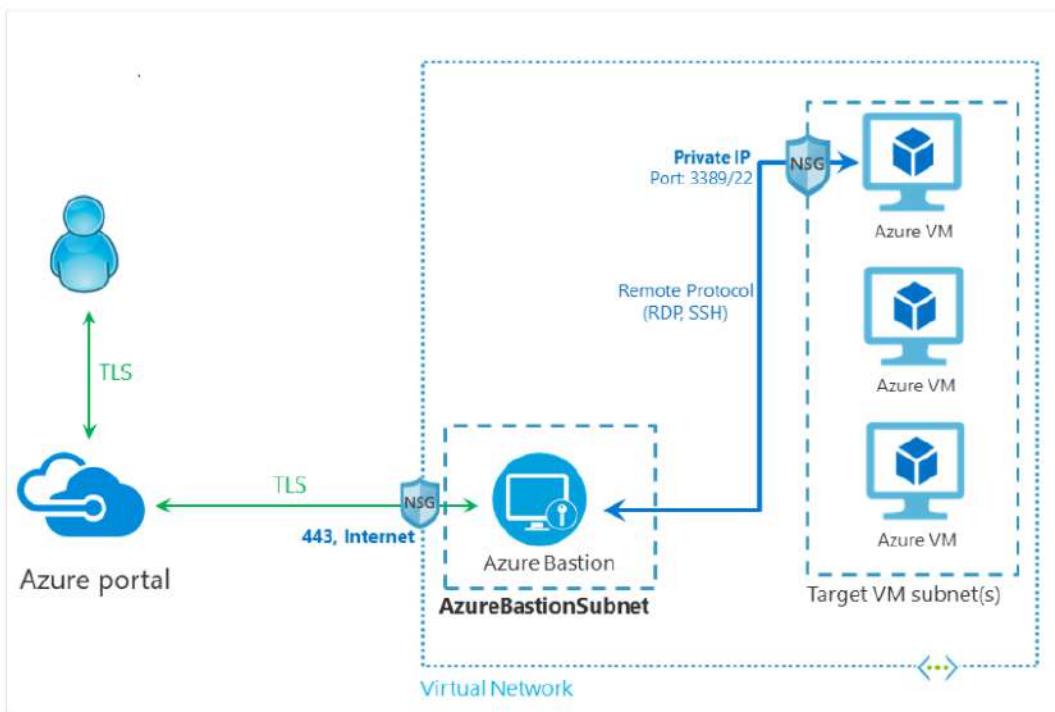
## Version feature comparison

For an Application Gateway v1-v2 feature comparison, see [Autoscaling and Zone-redundant Application Gateway v2](#).

# Azure Bastion

Azure Bastion is a service you deploy that lets you connect to a virtual machine using your browser and the Azure portal. The Azure Bastion service is a fully platform-managed PaaS service that you provision inside your virtual network. It provides secure and seamless RDP/SSH connectivity to your virtual machines directly from the Azure portal over TLS. When you connect via Azure Bastion, your virtual machines do not need a public IP address, agent, or special client software.

Bastion provides secure RDP and SSH connectivity to all of the VMs in the virtual network in which it is provisioned. Using Azure Bastion protects your virtual machines from exposing RDP/SSH ports to the outside world, while still providing secure access using RDP/SSH.



## Key benefits

- **RDP and SSH directly in Azure portal:** You can get to the RDP and SSH session directly in the Azure portal using a single click seamless experience.
- **Remote Session over TLS and firewall traversal for RDP/SSH:** Azure Bastion uses an HTML5 based web client that is automatically streamed to your local device. You get your RDP/SSH session over TLS on port 443, enabling you to traverse corporate firewalls securely.
- **No Public IP required on the Azure VM:** Azure Bastion opens the RDP/SSH connection to your Azure virtual machine using private IP on your VM. You don't need a public IP on your virtual machine.

- No hassle of managing Network Security Groups (NSGs): Azure Bastion is a fully managed platform PaaS service from Azure that is hardened internally to provide you secure RDP/SSH connectivity. You don't need to apply any NSGs to the Azure Bastion subnet. Because Azure Bastion connects to your virtual machines over private IP, you can configure your NSGs to allow RDP/SSH from Azure Bastion only. This removes the hassle of managing NSGs each time you need to securely connect to your virtual machines. For more information about NSGs, see [Network Security Groups](#).
- Protection against port scanning: Because you do not need to expose your virtual machines to the public Internet, your VMs are protected against port scanning by rogue and malicious users located outside your virtual network.
- Protect against zero-day exploits. Hardening in one place only: Azure Bastion is a fully platform-managed PaaS service. Because it sits at the perimeter of your virtual network, you don't need to worry about hardening each of the virtual machines in your virtual network. The Azure platform protects against zero-day exploits by keeping the Azure Bastion hardened and always up to date for you.

## SKUs

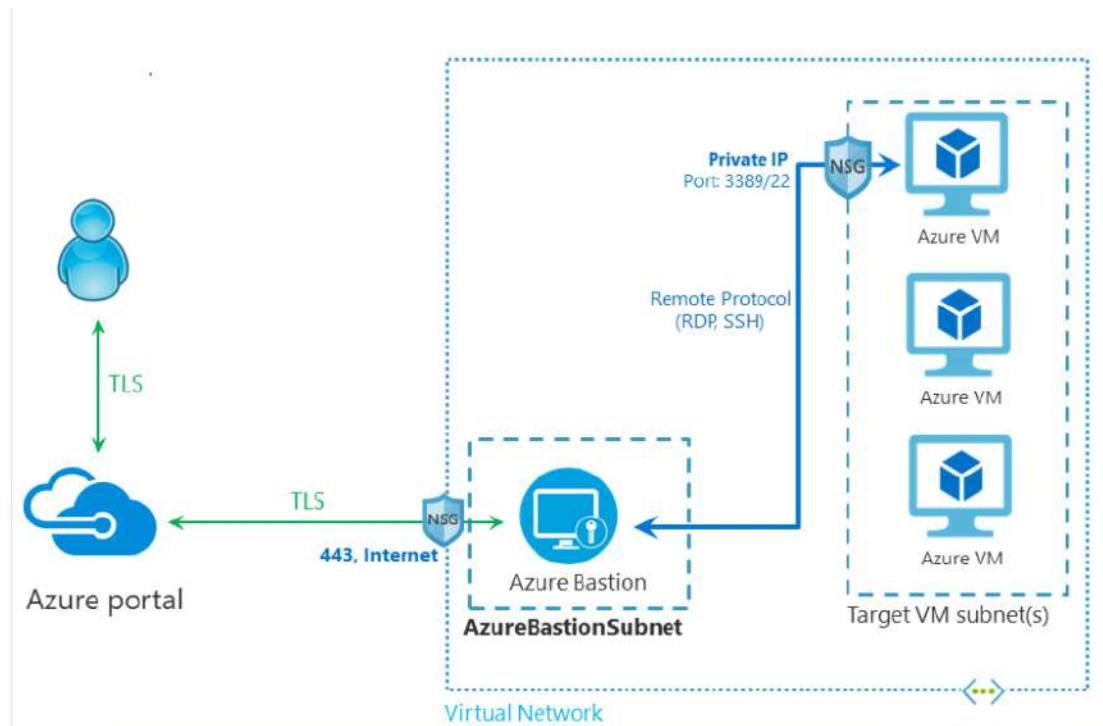
Azure Bastion has two available SKUs, Basic and Standard. For more information, including how to upgrade a SKU, see the [Configuration settings](#) article. The following table shows features and corresponding SKUs.

SKUs		
Feature	Basic SKU	Standard SKU
Connect to target VMs in peered virtual networks	Available	Available
Access Linux VM Private Keys in Azure Key Vault (AKV)	Available	Available
Host scaling	N/A	Available
Specify custom inbound port	N/A	Available
Connect to Linux VM using RDP	N/A	Available
Connect to Windows VM using SSH	N/A	Available

## Architecture

Azure Bastion is deployed to a virtual network and supports virtual network peering. Specifically, Azure Bastion manages RDP/SSH connectivity to VMs created in the local or peered virtual networks.

RDP and SSH are some of the fundamental means through which you can connect to your workloads running in Azure. Exposing RDP/SSH ports over the Internet isn't desired and is seen as a significant threat surface. This is often due to protocol vulnerabilities. To contain this threat surface, you can deploy bastion hosts (also known as jump-servers) at the public side of your perimeter network. Bastion host servers are designed and configured to withstand attacks. Bastion servers also provide RDP and SSH connectivity to the workloads sitting behind the bastion, as well as further inside the network.



This figure shows the architecture of an Azure Bastion deployment. In this diagram:

- The Bastion host is deployed in the virtual network that contains the AzureBastionSubnet subnet that has a minimum /26 prefix.
- The user connects to the Azure portal using any HTML5 browser.
- The user selects the virtual machine to connect to.
- With a single click, the RDP/SSH session opens in the browser.
- No public IP is required on the Azure VM.

## Host Scaling

Azure Bastion supports manual host scaling. You can configure the number of host instances (scale units) in order to manage the number of concurrent RDP/SSH connections that Azure Bastion can support. Increasing the number of host instances lets Azure Bastion manage more concurrent sessions. Decreasing the number of instances decreases the number of concurrent supported sessions. Azure Bastion supports up to 50 host instances. This feature is available for the Azure Bastion Standard SKU only.

For more information, see the [Configuration settings](#) article.

## Pricing

Azure Bastion pricing involves a combination of hourly pricing based on SKU, scale units, and data transfer rates. Pricing information can be found on the [Pricing](#) page.

## Azure DNS

Azure DNS is a hosting service for DNS domains that provides name resolution by using Microsoft Azure infrastructure. By hosting your domains in Azure, you can manage your DNS records by using the same credentials, APIs, tools, and billing as your other Azure services.

You can't use Azure DNS to buy a domain name. For an annual fee, you can buy a domain name by using App Service domains or a third-party domain name registrar. Your domains then can be hosted in Azure DNS for record management. For more information, see [Delegate a domain to Azure DNS](#).

The following features are included with Azure DNS.

### Reliability and performance

DNS domains in Azure DNS are hosted on Azure's global network of DNS name servers. Azure DNS uses anycast networking. Each DNS query is answered by the closest available DNS server to provide fast performance and high availability for your domain.

## Security

Azure DNS is based on Azure Resource Manager, which provides features such as:

- Azure role-based access control (Azure RBAC) to control who has access to specific actions for your organization.
- Activity logs to monitor how a user in your organization modified a resource or to find an error when troubleshooting.
- Resource locking to lock a subscription, resource group, or resource. Locking prevents other users in your organization from accidentally deleting or modifying critical resources.

For more information, see [How to protect DNS zones and records](#).

## DNSSEC

Azure DNS does not currently support DNSSEC. In most cases, you can reduce the need for DNSSEC by consistently using HTTPS/TLS in your applications. If DNSSEC is a critical requirement for your DNS zones, you can host these zones with third-party DNS hosting providers.

## Ease of use

Azure DNS can manage DNS records for your Azure services and provide DNS for your external resources as well. Azure DNS is integrated in the Azure portal and uses the same credentials, support contract, and billing as your other Azure services.

DNS billing is based on the number of DNS zones hosted in Azure and on the number of DNS queries received. To learn more about pricing, see [Azure DNS pricing](#).

Your domains and records can be managed by using the Azure portal, Azure PowerShell cmdlets, and the cross-platform Azure CLI. Applications that require automated DNS management can integrate with the service by using the REST API and SDKs.

## Customisable virtual networks with private domains

Azure DNS also supports private DNS domains. This feature allows you to use your own custom domain names in your private virtual networks rather than the Azure-provided names available today.

For more information, see [Use Azure DNS for private domains](#).

## Alias Records

Azure DNS supports alias record sets. You can use an alias record set to refer to an Azure resource, such as an Azure public IP address, an Azure Traffic Manager profile, or an Azure Content Delivery Network (CDN) endpoint. If the IP address of the underlying resource changes, the alias record set seamlessly updates itself during DNS resolution. The alias record set points to the service instance, and the service instance is associated with an IP address.

Also, you can now point your apex or naked domain to a Traffic Manager profile or CDN endpoint using an alias record. An example is contoso.com. For more information, see [Overview of Azure DNS alias records](#).

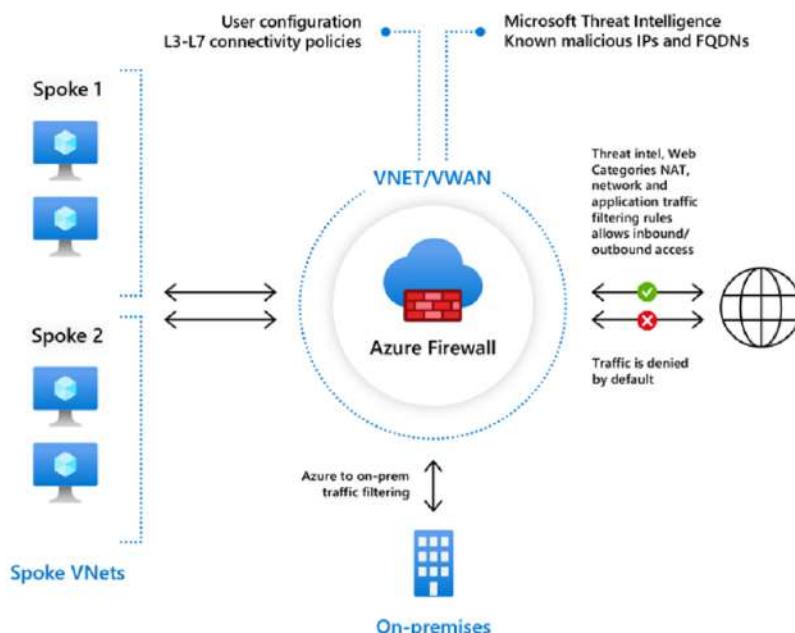
## Azure Firewall

Azure Firewall is a cloud-native and intelligent network firewall security service that provides the best of breed threat protection for your cloud workloads running in Azure. It's a fully stateful, firewall as a service with built-in high availability and unrestricted cloud scalability. It provides both east-west and north-south traffic inspection.

Azure Firewall is offered in two SKUs: Standard and Premium.

### Azure Firewall Standard

Azure Firewall Standard provides L3-L7 filtering and threat intelligence feeds directly from Microsoft Cyber Security. Threat intelligence-based filtering can alert and deny traffic from/to known malicious IP addresses and domains which are updated in real time to protect against new and emerging attacks.



## Azure Firewall Standard Features

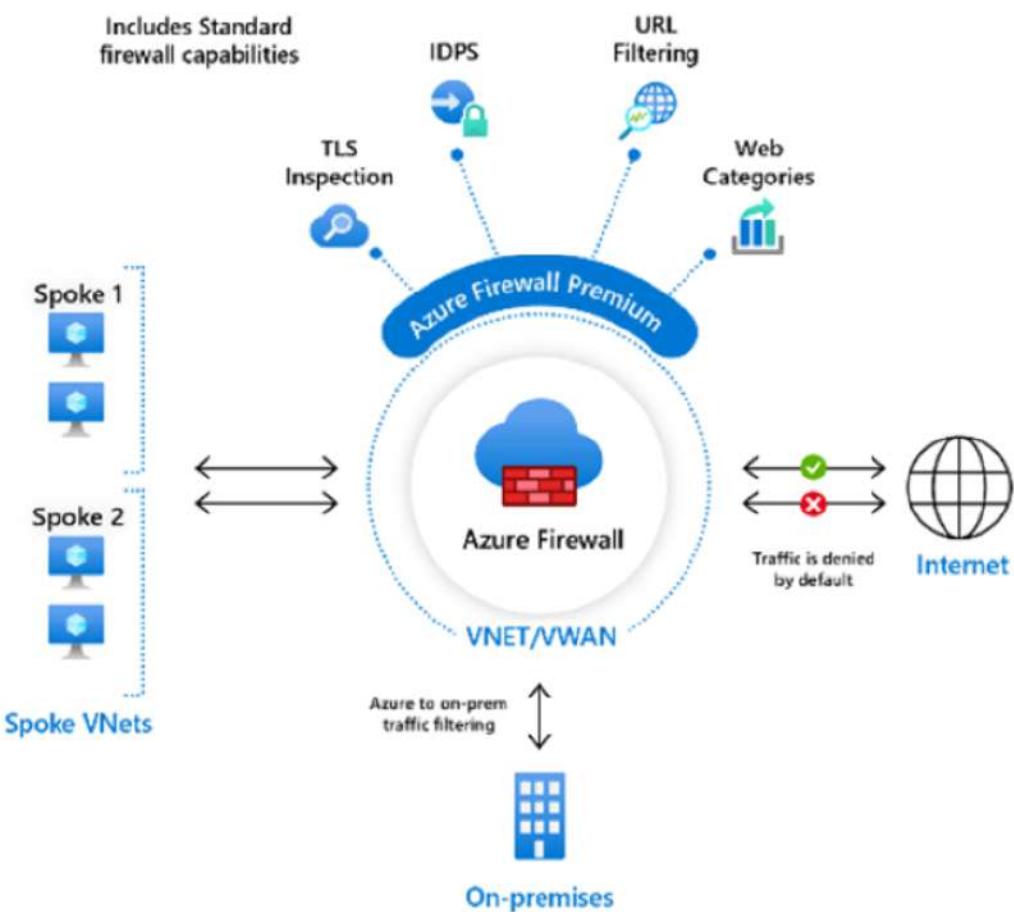
Azure Firewall includes the following features:

- Built-in high availability
- Availability Zones
- Unrestricted cloud scalability
- Application FQDN filtering rules
- Network traffic filtering rules
- FQDN tags
- Service tags
- Threat intelligence
- DNS proxy
- Custom DNS
- FQDN in network rules
- Deployment without public IP address in Forced Tunnel Mode
- Outbound SNAT support
- Inbound DNAT support
- Multiple public IP addresses
- Azure Monitor logging
- Forced tunneling
- Web categories
- Certifications

## Azure Firewall Premium

Azure Firewall Premium provides advanced capabilities include signature-based IDPS to allow rapid detection of attacks by looking for specific patterns. These patterns can include byte sequences in network traffic, or known malicious instruction sequences used by malware.

There are more than 58,000 signatures in over 50 categories which are updated in real time to protect against new and emerging exploits. The exploit categories include malware, phishing, coin mining, and Trojan attacks.



## Azure Firewall Premium Features

Azure Firewall Premium includes the following features:

- TLS inspection - decrypts outbound traffic, processes the data, then encrypts the data and sends it to the destination.
- IDP - A network intrusion detection and prevention system (IDPS) allows you to monitor network activities for malicious activity, log information about this activity, report it, and optionally attempt to block it.
- URL filtering - extends Azure Firewall's FQDN filtering capability to consider an entire URL. For example, www.contoso.com/a/c instead of www.contoso.com.
- Web categories - administrators can allow or deny user access to website categories such as gambling websites, social media websites, and others.

## Azure Firewall Manager

Azure Firewall Manager is a security management service that provides central security policy and route management for cloud-based security perimeters.

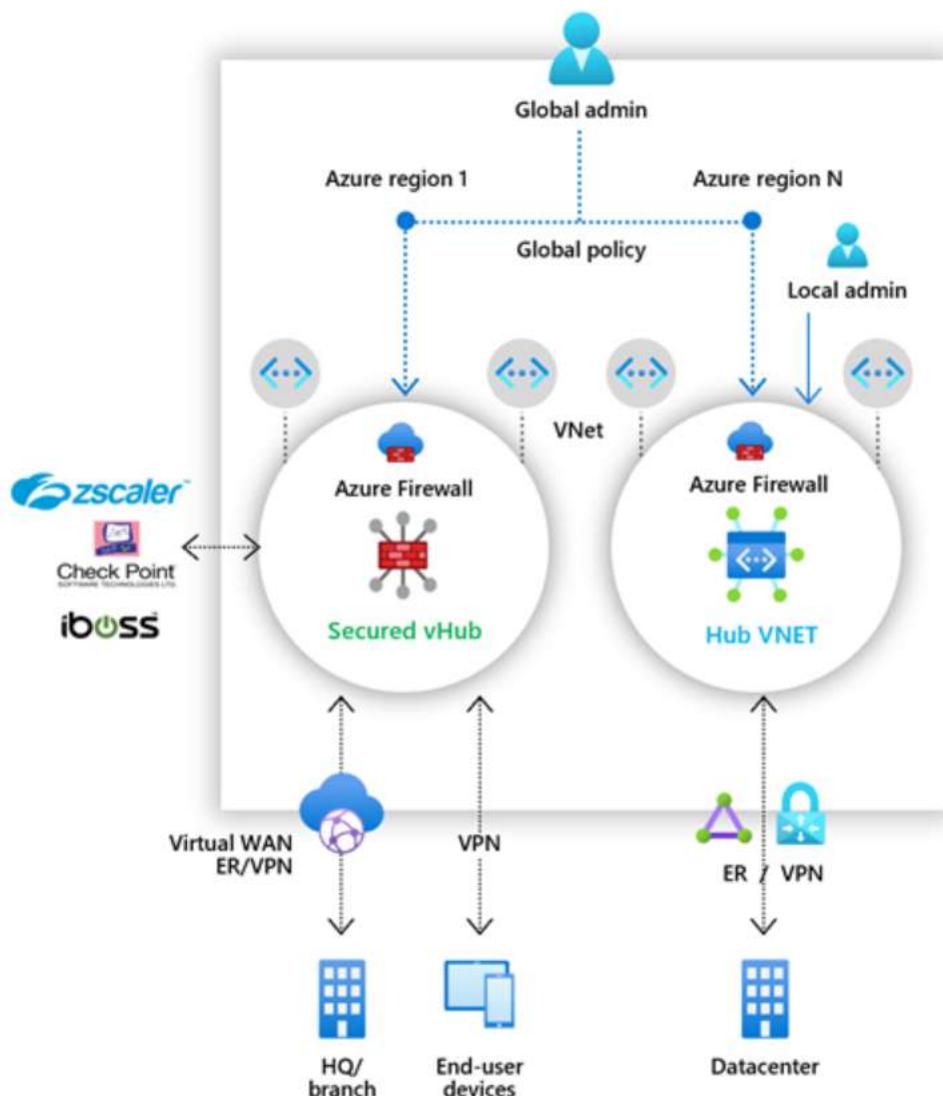
Firewall Manager can provide security management for two network architecture types:

- Secured virtual hub

An Azure Virtual WAN Hub is a Microsoft-managed resource that lets you easily create hub and spoke architectures. When security and routing policies are associated with such a hub, it is referred to as a secured virtual hub.

- Hub virtual network

This is a standard Azure virtual network that you create and manage yourself. When security policies are associated with such a hub, it is referred to as a hub virtual network. At this time, only Azure Firewall Policy is supported. You can peer spoke virtual networks that contain your workload servers and services. You can also manage firewalls in standalone virtual networks that aren't peered to any spoke.



## Azure Firewall Manager Features

Azure Firewall Manager offers the following features:

- Central Azure Firewall deployment and configuration
  - You can centrally deploy and configure multiple Azure Firewall instances that span different Azure regions and subscriptions.
- Hierarchical policies (global and local)
  - You can use Azure Firewall Manager to centrally manage Azure Firewall policies across multiple secured virtual hubs. Your central IT teams can author global firewall policies to enforce organization wide firewall policy across teams. Locally authored firewall policies allow a DevOps self-service model for better agility.
- Integrated with third-party security-as-a-service for advanced security
  - In addition to Azure Firewall, you can integrate third-party security as a service (SECaaS) providers to provide additional network protection for your VNet and branch Internet connections.

This feature is available only with secured virtual hub deployments.

- VNet to Internet (V2I) traffic filtering
  - Filter outbound virtual network traffic with your preferred third-party security provider.
  - Leverage advanced user-aware Internet protection for your cloud workloads running on Azure.
- Branch to Internet (B2I) traffic filtering
  - Leverage your Azure connectivity and global distribution to easily add third-party filtering for branch to Internet scenarios.

## Azure Front Door

Azure Front Door is a global, scalable entry-point that uses the Microsoft global edge network to create fast, secure, and widely scalable web applications. With Front Door, you can transform your global consumer and enterprise applications into robust, high-performing personalized modern applications with contents that reach a global audience through Azure.

Front Door works at Layer 7 (HTTP/HTTPS layer) using anycast protocol with split TCP and Microsoft's global network to improve global connectivity. Based on your routing method you can ensure that Front Door will route your client requests to the fastest and most available application backend. An application backend is any Internet-facing service hosted inside or outside of Azure. Front Door provides a range of traffic-routing methods and backend health monitoring options to suit different application needs and automatic failover scenarios. Similar to Traffic Manager, Front Door is resilient to failures, including failures to an entire Azure region.

## Why use Azure Front Door?

With Front Door you can build, operate, and scale out your dynamic web application and static content. Front Door enables you to define, manage, and monitor the global routing for your web traffic by optimizing for top-tier end-user performance and reliability through quick global failover.

Key features included with Front Door:

- Accelerated application performance by using split TCP-based anycast protocol.
- Intelligent health probe monitoring for backend resources.
- URL-path based routing for requests.
- Enables hosting of multiple websites for efficient application infrastructure.
- Cookie-based session affinity.
- SSL offloading and certificate management.
- Define your own custom domain.
- Application security with integrated Web Application Firewall (WAF).
- Redirect HTTP traffic to HTTPS with URL redirect.
- Custom forwarding path with URL rewrite.
- Native support of end-to-end IPv6 connectivity and HTTP/2 protocol.

## Azure Orbital

Azure Orbital is a fully managed cloud-based ground station as a service that allows you to streamline your operations by ingesting space data directly into Azure. With Azure Orbital, you can focus on your missions by off-loading the responsibility for deployment and maintenance of ground stations.

Azure Orbital uses Microsoft's global infrastructure and low-latency global network along with an expansive partner ecosystem of ground station networks, cloud modems, and "Telemetry, Tracking, & Control" (TT&C) functions.



Azure Orbital offers two main services:

- Azure Orbital Earth Observation
  - Schedule contacts with satellites on a pay-as-you-go basis to ingest data from the satellite, monitor the satellite health and status, or transmit commands to the satellite. Incoming data is delivered to your private virtual network allowing it to be processed or stored in Azure.

The fully digitized service allows you to use software modems from Kratos and Amerigint to do the modulation / demodulation, and encoding / decoding functions to recover the data.

For a full end-to-end solution to manage fleet operations and "Telemetry, Tracking, & Control" (TT&C) functions, seamlessly integrate your Azure Orbital operations with Kubos Major Tom. Lower your operational costs and maximize your capabilities by using Azure Space.

- Spacecraft contact self-service scheduling
- Direct data ingestion into Azure
- Marketplace integration with third-party data processing and image calibration services
- Integrated cloud modems for X and S bands and Certified cloud modems available through the Azure Marketplace
- Global reach through integrated third-party networks
- Azure Orbital Global Communications
- Satellite operators who provide global communication capabilities to their customers can route their traffic through the Microsoft global network.

They can offer private connection to their customer's virtual network, or offer other managed services to their customers by connecting them to the operator's virtual network.

In addition, all internet traffic destined to Microsoft services (including Office365, Microsoft Teams, Xbox, Azure public IPs) can be routed directly within region and without traversing an ISP. It can reduce the amount of traffic going towards the internet and provide lower latency access to these services.

Operators can colocate new ground stations at Azure data centers or at Azure Edges, or inter-connect existing ground stations with the global Azure backbone.

Azure Orbital delivers the traffic from an Orbital ground station to your virtual network, enabling you to bundle and provide managed security and connectivity services to your end-customers.

- Routing over global Microsoft network
- Internet breakout at the edge
- Traffic delivery to provider's virtual network
- Service chain other Azure services to provide managed services
- Private connection to customer's virtual network

## Express Route

To connect to Microsoft cloud services using ExpressRoute, you'll need to set up and manage routing. Some connectivity providers offer setting up and managing routing as a managed service. Check with your connectivity provider to see if they offer this service. If they don't, you must adhere to the following requirements:  
Refer to the Circuits and routing domains article for a description of the routing sessions that need to be set up in to facilitate connectivity.

### IP addresses used for peerings

You need to reserve a few blocks of IP addresses to configure routing between your network and Microsoft's Enterprise edge (MSEEs) routers. This section provides a list of requirements and describes the rules regarding how these IP addresses must be acquired and used.

#### IP addresses used for Azure private peering

You can use either private IP addresses or public IP addresses to configure the peerings. The address range used for configuring routes must not overlap with address ranges used to create virtual networks in Azure.

- **IPv4:**

- You must reserve a /29 subnet or two /30 subnets for routing interfaces.
- The subnets used for routing can be either private IP addresses or public IP addresses.
- The subnets must not conflict with the range reserved by the customer for use in the Microsoft cloud.
- If a /29 subnet is used, it is split into two /30 subnets.
- The first /30 subnet is used for the primary link and the second /30 subnet is used for the secondary link.
- For each of the /30 subnets, you must use the first IP address of the /30 subnet on your router. Microsoft uses the second IP address of the /30 subnet to set up a BGP session.
- You must set up both BGP sessions for our availability SLA to be valid.

- **IPv6:**

- You must reserve a /125 subnet or two /126 subnets for routing interfaces.
- The subnets used for routing can be either private IP addresses or public IP addresses.
- The subnets must not conflict with the range reserved by the customer for use in the Microsoft cloud.
- If a /125 subnet is used, it is split into two /126 subnets.
- The first /126 subnet is used for the primary link and the second /126 subnet is used for the secondary link.
- For each of the /126 subnets, you must use the first IP address of the /126 subnet on your router. Microsoft uses the second IP address of the /126 subnet to set up a BGP session.
- You must set up both BGP sessions for our availability SLA to be valid.

## **Example for private peering**

If you choose to use a.b.c.d/29 to set up the peering, it is split into two /30 subnets. In the following example, notice how the a.b.c.d/29 subnet is used:

- a.b.c.d/29 is split to a.b.c.d/30 and a.b.c.d+4/30 and passed down to Microsoft through the provisioning APIs.
- You use a.b.c.d+1 as the VRF IP for the Primary PE and Microsoft will consume a.b.c.d+2 as the VRF IP for the primary MSEE.
- You use a.b.c.d+5 as the VRF IP for the secondary PE and Microsoft will use a.b.c.d+6 as the VRF IP for the secondary MSEE.

Consider a case where you select 192.168.100.128/29 to set up private peering. 192.168.100.128/29 includes addresses from 192.168.100.128 to 192.168.100.135, among which:

- 192.168.100.128/30 will be assigned to link1, with provider using 192.168.100.129 and Microsoft using 192.168.100.130.
- 192.168.100.132/30 will be assigned to link2, with provider using 192.168.100.133 and Microsoft using 192.168.100.134.

## IP addresses used for Microsoft peering

You must use public IP addresses that you own for setting up the BGP sessions. Microsoft must be able to verify the ownership of the IP addresses through Routing Internet Registries and Internet Routing Registries.

- The IPs listed in the portal for Advertised Public Prefixes for Microsoft Peering will create ACLs for the Microsoft core routers to allow inbound traffic from these IPs.
- You must use a unique /29 (IPv4) or /125 (IPv6) subnet or two /30 (IPv4) or /126 (IPv6) subnets to set up the BGP peering for each peering per ExpressRoute circuit (if you have more than one).
- If a /29 subnet is used, it is split into two /30 subnets.
- The first /30 subnet is used for the primary link and the second /30 subnet will be used for the secondary link.
- For each of the /30 subnets, you must use the first IP address of the /30 subnet on your router. Microsoft uses the second IP address of the /30 subnet to set up a BGP session.
- If a /125 subnet is used, it is split into two /126 subnets.
- The first /126 subnet is used for the primary link and the second /126 subnet will be used for the secondary link.
- For each of the /126 subnets, you must use the first IP address of the /126 subnet on your router. Microsoft uses the second IP address of the /126 subnet to set up a BGP session.
- You must set up both BGP sessions for our availability SLA to be valid.

## IP addresses used for Azure public peering

You must use public IP addresses that you own for setting up the BGP sessions. Microsoft must be able to verify the ownership of the IP addresses through Routing Internet Registries and Internet Routing Registries.

- You must use a unique /29 subnet or two /30 subnets to set up the BGP peering for each peering per ExpressRoute circuit (if you have more than one).
- If a /29 subnet is used, it is split into two /30 subnets.
- The first /30 subnet is used for the primary link and the second /30 subnet is used for the secondary link.

- For each of the /30 subnets, you must use the first IP address of the /30 subnet on your router. Microsoft uses the second IP address of the /30 subnet to set up a BGP session.
- You must set up both BGP sessions for our availability SLA to be valid.

## **Public IP address requirement**

### **Private peering**

You can choose to use public or private IPv4 addresses for private peering. We provide end-to-end isolation of your traffic, so overlapping of addresses with other customers is not possible in case of private peering. These addresses are not advertised to Internet.

### **Microsoft peering**

The Microsoft peering path lets you connect to Microsoft cloud services. The list of services includes Microsoft 365 services, such as Exchange Online, SharePoint Online, Skype for Business, and Microsoft Teams. Microsoft supports bi-directional connectivity on the Microsoft peering. Traffic destined to Microsoft cloud services must use valid public IPv4 addresses before they enter the Microsoft network.

Make sure that your IP address and AS number are registered to you in one of the following registries:

- ARIN
- APNIC
- AFRINIC
- LACNIC
- RIPE NCC
- RADB
- ALTDB

If your prefixes and AS number are not assigned to you in the preceding registries, you need to open a support case for manual validation of your prefixes and ASN. Support requires documentation, such as a Letter of Authorization, that proves you are allowed to use the resources.

A Private AS Number is allowed with Microsoft Peering, but will also require manual validation. In addition, we remove private AS numbers in the AS PATH for the received prefixes. As a result, you can't append private AS numbers in the AS PATH to influence routing for Microsoft Peering. Additionally, AS numbers 64496 - 64511 reserved by IANA for documentation purposes are not allowed in the path.

Public peering (deprecated - not available for new circuits)

The Azure public peering path enables you to connect to all services hosted in Azure over their public IP addresses. These include services listed in the ExpressRoute FAQ and any services hosted by ISVs on Microsoft Azure. Connectivity to Microsoft Azure services on public peering is always initiated from your network into the Microsoft network. You must use Public IP addresses for the traffic destined to Microsoft network.

### **Dynamic route exchange**

Routing exchange will be over eBGP protocol. EBGP sessions are established between the MSEEs and your routers. Authentication of BGP sessions is not a requirement. If required, an MD5 hash can be configured. See the Configure routing and Circuit provisioning workflows and circuit states for information about configuring BGP sessions.

### **Autonomous System numbers**

Microsoft uses AS 12076 for Azure public, Azure private and Microsoft peering. We have reserved ASNs from 65515 to 65520 for internal use. Both 16 and 32 bit AS numbers are supported.

There are no requirements around data transfer symmetry. The forward and return paths may traverse different router pairs. Identical routes must be advertised from either sides across multiple circuit pairs belonging to you. Route metrics are not required to be identical.

### **Route aggregation and prefix limits**

We support up to 4000 IPv4 prefixes and 100 IPv6 prefixes advertised to us through the Azure private peering. This can be increased up to 10,000 IPv4 prefixes if the ExpressRoute premium add-on is enabled. We accept up to 200 prefixes per BGP session for Azure public and Microsoft peering.

The BGP session is dropped if the number of prefixes exceeds the limit. We will accept default routes on the private peering link only. Provider must filter out default route and private IP addresses (RFC 1918) from the Azure public and Microsoft peering paths.

### **Transit routing and cross-region routing**

ExpressRoute cannot be configured as transit routers. You will have to rely on your connectivity provider for transit routing services.

## **Advertising default routes**

Default routes are permitted only on Azure private peering sessions. In such a case, we will route all traffic from the associated virtual networks to your network. Advertising default routes into private peering will result in the internet path from Azure being blocked. You must rely on your corporate edge to route traffic from and to the internet for services hosted in Azure.

To enable connectivity to other Azure services and infrastructure services, you must make sure one of the following items is in place:

- Azure public peering is enabled to route traffic to public endpoints.
- You use user-defined routing to allow internet connectivity for every subnet requiring Internet connectivity.

## **Support for BGP communities**

This section provides an overview of how BGP communities will be used with ExpressRoute. Microsoft will advertise routes in the private, Microsoft and public (deprecated) peering paths with routes tagged with appropriate community values. The rationale for doing so and the details on community values are described below. Microsoft, however, will not honor any community values tagged to routes advertised to Microsoft.

For private peering, if you configure a custom BGP community value on your Azure virtual networks, you will see this custom value and a regional BGP community value on the Azure routes advertised to your on-premises over ExpressRoute.

For Microsoft peering, you are connecting to Microsoft through ExpressRoute at any one peering location within a geopolitical region, you will have access to all Microsoft cloud services across all regions within the geopolitical boundary.

For example, if you connected to Microsoft in Amsterdam through ExpressRoute, you will have access to all Microsoft cloud services hosted in North Europe and West Europe.

Refer to the ExpressRoute partners and peering locations page for a detailed list of geopolitical regions, associated Azure regions, and corresponding ExpressRoute peering locations.

You can purchase more than one ExpressRoute circuit per geopolitical region. Having multiple connections offers you significant benefits on high availability due to geo-redundancy. In cases where you have multiple ExpressRoute circuits, you will receive the same set of prefixes advertised from Microsoft on the Microsoft peering

and public peering paths. This means you will have multiple paths from your network into Microsoft. This can potentially cause suboptimal routing decisions to be made within your network. As a result, you may experience suboptimal connectivity experiences to different services. You can rely on the community values to make appropriate routing decisions to offer optimal routing to users.

## Service to BGP community value

In addition to the above, Microsoft will also tag prefixes based on the service they belong to. This applies only to the Microsoft peering. The table below provides a mapping of service to BGP community value. You can run the 'Get-AzBgpServiceCommunity' cmdlet for a full list of the latest values.

Service	BGP community value
Exchange Online**	12076:5010
SharePoint Online**	12076:5020
Skype For Business Online**/***	12076:5030
CRM Online****	12076:5040
Azure Global Services*	12076:5050
Azure Active Directory	12076:5060
Azure Resource Manager	12076:5070
Other Office 365 Online services**	12076:5100

\* Azure Global Services includes only Azure DevOps at this time.

\*\* Authorization required from Microsoft, refer Configure route filters for Microsoft Peering

\*\*\* This community also publishes the needed routes for Microsoft Teams services.

\*\*\*\* CRM Online supports Dynamics v8.2 and below. For higher versions, select the regional community for your Dynamics deployments.

## BGP Community support in National Clouds

BGP COMMUNITY SUPPORT IN NATIONAL CLOUDS	
National Clouds Azure Region	BGP community value
US Government	
US Gov Arizona	12076:51106
US Gov Iowa	12076:51109
US Gov Virginia	12076:51105
US Gov Texas	12076:51108
US DoD Central	12076:51209
US DoD East	12076:51205
China	
China North	12076:51301
China East	12076:51302
China East 2	12076:51303
China North 2	12076:51304
BGP COMMUNITY SUPPORT IN NATIONAL CLOUDS	
Service in National Clouds	BGP community value
US Government	
Exchange Online	12076:5110
SharePoint Online	12076:5120
Skype For Business Online	12076:5130
Azure Active Directory	12076:5160
Other Office 365 Online services	12076:5200

- Office 365 communities are not supported over Microsoft Peering for Azure China region.

# Azure Internet Analyzer

Internet Analyzer is a client-side measurement platform to test how networking infrastructure changes impact your customers' performance. Whether you're migrating from on-premises to Azure or evaluating a new Azure service, Internet Analyzer allows you to learn from your users' data and Microsoft's rich analytics to better understand and optimize your network architecture with Azure—before you migrate.

Internet Analyzer uses a small JavaScript client embedded in your Web application to measure the latency from your end users to your selected set of network destinations, we call endpoints. Internet Analyzer allows you to set up multiple side-by-side tests, allowing you to evaluate a variety of scenarios as your infrastructure and customer needs evolves. Internet Analyzer provides custom and preconfigured endpoints, providing you both the convenience and flexibility to make trusted performance decisions for your end users.

## Quick & customizable tests

Internet Analyzer addresses performance-related questions for cloud migration, deploying to a new or additional Azure regions, or testing new application and content delivery platforms in Azure, such as Azure Front Door and Microsoft Azure CDN.

Each test you create in Internet Analyzer is composed of two endpoints—Endpoint A and Endpoint B. Endpoint B's performance is analyzed relative to Endpoint A.

You can either configure your own custom endpoint or select from a variety of preconfigured Azure endpoints. Custom endpoints should be used to evaluate on-premises workloads, your instances in other cloud providers, or your custom Azure configurations. Tests may be composed of two custom endpoints; however, at least one custom endpoint must be hosted in Azure. Preconfigured Azure endpoints are a quick and easy way to evaluate the performance of popular Azure networking platforms such as Azure Front Door, Azure Traffic Manager, and Azure CDN.

During preview, the following preconfigured endpoints are available:

- Azure regions
  - Brazil South
  - Central India

- Central US
- East Asia
- East US
- Japan West
- North Europe
- South Africa North
- Southeast Asia
- UAE North
- UK West
- West Europe
- West US
- West US 2

➤ Multiple Azure region combinations

- East US, Brazil South
- East US, East Asia
- West Europe, Brazil South
- West Europe, Southeast Asia
- West Europe, UAE North
- West US, East US
- West US, West Europe
- West US, UAE North
- West Europe, UAE North, Southeast Asia
- West US, West Europe, East Asia
- West US, North Europe, Southeast Asia, UAE North, South Africa North

➤ Azure + Azure Front Door - deployed on any single or multiple Azure region combinations listed above

➤ Azure + Azure CDN from Microsoft - deployed on any single Azure region combination listed above

➤ Azure + Azure Traffic Manager - deployed on any multiple Azure region combination listed above

## Suggested test scenarios

To help you make the best performance decisions for your customers, Internet Analyzer allows you to evaluate two endpoints for your specific population of end users.

While Internet Analyzer can answer a multitude of questions, some of the most common are:

- What is the performance impact of migrating to the cloud?

Suggested Test: Custom (your current on-premises infrastructure) vs. Azure (any preconfigured endpoint)

- What is the value of putting my data at the edge vs. in a data center?

Suggested Test: Azure vs. Azure Front Door, Azure vs. Azure CDN from Microsoft

- What is the performance benefit of Azure Front Door?

Suggested Test: Custom/ Azure/ CDN vs. Azure Front Door

- What is the performance benefit of Azure CDN from Microsoft?

Suggested Test: Custom/ Azure/ AFD vs. Azure CDN from Microsoft

- How does Azure CDN from Microsoft stack up?

Suggested Test: Custom (other CDN endpoint) vs. Azure CDN from Microsoft

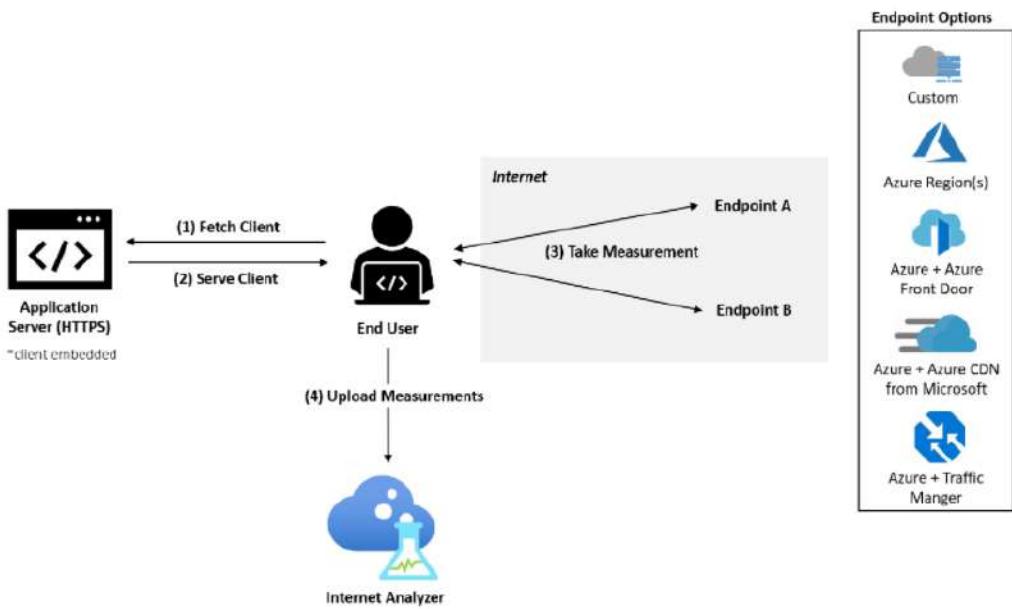
- What is the best cloud for your end-user population in each region?

Suggested Test: Custom (other cloud service) vs. Azure (any preconfigured endpoint)

## How it works

To use Internet Analyzer, set up an Internet Analyzer resource in the Microsoft Azure portal and install the small JavaScript client in your application. The client measures the latency from your end users to your selected endpoints by downloading a one-pixel image over HTTPS. After collecting latency measurements, the client sends the measurement data to Internet Analyzer.

When a user visits the Web application, the JavaScript client selects two endpoints to measure across all configured tests. For each endpoint, the client performs a cold and warm measurement. The cold measurement incurs additional latency beside the pure network latency between the user and endpoint such as DNS resolution, TCP connection handshake, and SSL/TLS negotiation. The warm measurement follows just after the cold measurement completes and takes advantage of modern browsers' persistent TCP connection management to get an accurate measure of end-to-end latency. When supported by the user's browser, the W3C resource timing API is used for accurate measurement timing. Currently, only warm latency measurements are used for analysis.



## Scorecards

Once a test starts, telemetry data is visible in your Internet Analyzer resource under the Scorecard tab. This data is always aggregated. Use the following filters to change which view of the data you see:

- **Test:** Select the test that you'd like to view results for. Test data appears once there is enough data to complete the analysis – in most cases, within 24 hours.
- **Time period & end date:** Internet Analyzer generates three scorecards daily – each scorecard reflects a different aggregation time period – the 24 hours prior (day), the seven days prior (week), and the 30 days prior (month). Use the “End Date” filter to select the time period you want to see.
- **Country:** Use this filter to view data specific to end users residing in a country. The global filter shows data across all geographies.

# Azure Load Balancer

Azure Load Balancer operates at layer 4 of the Open Systems Interconnection (OSI) model. It's the single point of contact for clients. Load balancer distributes inbound flows that arrive at the load balancer's front end to backend pool instances. These flows are according to configured load-balancing rules and health probes. The backend pool instances can be Azure Virtual Machines or instances in a virtual machine scale set.

A public load balancer can provide outbound connections for virtual machines (VMs) inside your virtual network. These connections are accomplished by translating their private IP addresses to public IP addresses. Public Load Balancers are used to load balance internet traffic to your VMs.

An internal (or private) load balancer is used where private IPs are needed at the frontend only. Internal load balancers are used to load balance traffic inside a virtual network. A load balancer frontend can be accessed from an on-premises network in a hybrid scenario.

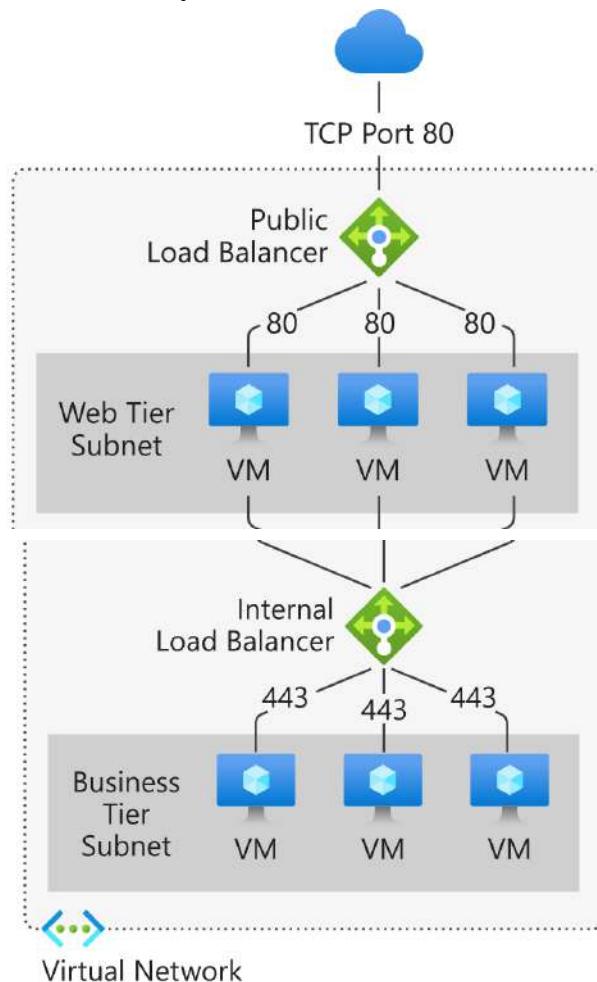


Figure: Balancing multi-tier applications by using both public and internal Load Balancer

## Why use Azure Load Balancer?

With Azure Load Balancer, you can scale your applications and create highly available services. Load balancer supports both inbound and outbound scenarios. Load balancer provides low latency and high throughput, and scales up to millions of flows for all TCP and UDP applications.

Key scenarios that you can accomplish using Azure Standard Load Balancer include:

- Load balance internal and external traffic to Azure virtual machines.
- Increase availability by distributing resources within and across zones.
- Configure outbound connectivity for Azure virtual machines.
- Use health probes to monitor load-balanced resources.
- Employ port forwarding to access virtual machines in a virtual network by public IP address and port.
- Enable support for load-balancing of IPv6.
- Standard load balancer provides multi-dimensional metrics through Azure Monitor. These metrics can be filtered, grouped, and broken out for a given dimension. They provide current and historic insights into performance and health of your service. Insights for Azure Load Balancer offers a preconfigured dashboard with useful visualizations for these metrics. Resource Health is also supported. Review Standard load balancer diagnostics for more details.
- Load balance services on multiple ports, multiple IP addresses, or both.
- Move internal and external load balancer resources across Azure regions.
- Load balance TCP and UDP flow on all ports simultaneously using HA ports.

## Secure by default

- Standard load balancer is built on the zero trust network security model.
- Standard Load Balancer is secure by default and part of your virtual network. The virtual network is a private and isolated network.
- Standard load balancers and standard public IP addresses are closed to inbound connections unless opened by Network Security Groups. NSGs are used to explicitly permit allowed traffic. If you don't have an NSG on a subnet or NIC of your virtual machine resource, traffic isn't allowed to reach this resource. To learn about NSGs and how to apply them to your scenario, see Network Security Groups.
- Basic load balancer is open to the internet by default.
- Load balancer doesn't store customer data.

# Azure Network Watcher

Azure Network Watcher provides tools to monitor, diagnose, view metrics, and enable or disable logs for resources in an Azure virtual network. Network Watcher is designed to monitor and repair the network health of IaaS (Infrastructure-as-a-Service) products which includes Virtual Machines, Virtual Networks, Application Gateways, Load balancers, etc. Note: It is not intended for and will not work for PaaS monitoring or Web analytics.

## Monitoring

### Monitor communication between a virtual machine and an endpoint

Endpoints can be another virtual machine (VM), a fully qualified domain name (FQDN), a uniform resource identifier (URI), or IPv4 address. The connection monitor capability monitors communication at a regular interval and informs you of reachability, latency, and network topology changes between the VM and the endpoint. For example, you might have a web server VM that communicates with a database server VM. Someone in your organization may, unknown to you, apply a custom route or network security rule to the web server or database server VM or subnet.

If an endpoint becomes unreachable, connection troubleshoot informs you of the reason. Potential reasons are a DNS name resolution problem, the CPU, memory, or firewall within the operating system of a VM, or the hop type of a custom route, or security rule for the VM or subnet of the outbound connection. Learn more about security rules and route hop types in Azure.

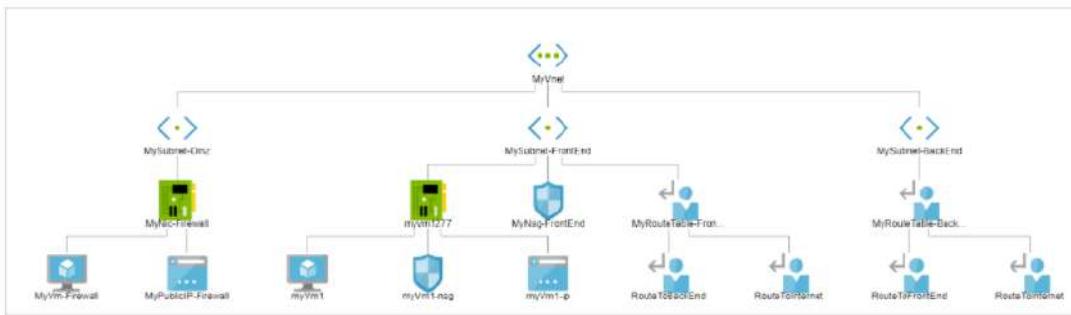
Connection monitor also provides the minimum, average, and maximum latency observed over time. After learning the latency for a connection, you may find that you're able to decrease the latency by moving your Azure resources to different Azure regions. Learn more about determining relative latencies between Azure regions and internet service providers and how to monitor communication between a VM and an endpoint with connection monitor. If you'd rather test a connection at a point in time, rather than monitor the connection over time, like you do with connection monitor, use the connection troubleshoot capability.

Network performance monitor is a cloud-based hybrid network monitoring solution that helps you monitor network performance between various points in your network infrastructure. It also helps you monitor network connectivity to service and application endpoints and monitor the performance of Azure ExpressRoute. Network performance monitor detects network issues like traffic blackholing, routing errors,

and issues that conventional network monitoring methods aren't able to detect. The solution generates alerts and notifies you when a threshold is breached for a network link. It also ensures timely detection of network performance issues and localizes the source of the problem to a particular network segment or device. Learn more about network performance monitor.

### View resources in a virtual network and their relationships

As resources are added to a virtual network, it can become difficult to understand what resources are in a virtual network and how they relate to each other. The topology capability enables you to generate a visual diagram of the resources in a virtual network, and the relationships between the resources. The following picture shows an example topology diagram for a virtual network that has three subnets, two VMs, network interfaces, public IP addresses, network security groups, route tables, and the relationships between the resources:



## Diagnostics

### Diagnose network traffic filtering problems to or from a VM

When you deploy a VM, Azure applies several default security rules to the VM that allow or deny traffic to or from the VM. You might override Azure's default rules, or create additional rules. At some point, a VM may become unable to communicate with other resources, because of a security rule. The IP flow verify capability enables you to specify a source and destination IPv4 address, port, protocol (TCP or UDP), and traffic direction (inbound or outbound). IP flow verify then tests the communication and informs you if the connection succeeds or fails. If the connection fails, IP flow verify tells you which security rule allowed or denied the communication, so that you can resolve the problem. Learn more about IP flow verify by completing the Diagnose a virtual machine network traffic filter problem tutorial.

### Diagnose network routing problems from a VM

When you create a virtual network, Azure creates several default outbound routes for network traffic. The outbound traffic from all resources, such as VMs, deployed in a virtual network, are routed based on Azure's default routes. You might override Azure's default routes, or create additional routes. You may find that a VM can no longer communicate with other resources because of a specific route. The next

hop capability enables you to specify a source and destination IPv4 address. Next hop then tests the communication and informs you what type of next hop is used to route the traffic. You can then remove, change, or add a route, to resolve a routing problem. Learn more about the next hop capability.

#### [Diagnose outbound connections from a VM](#)

The connection troubleshoot capability enables you to test a connection between a VM and another VM, an FQDN, a URI, or an IPv4 address. The test returns similar information returned when using the connection monitor capability, but tests the connection at a point in time, rather than monitoring it over time, as connection monitor does. Learn more about how to troubleshoot connections using connection-troubleshoot.

#### [Capture packets to and from a VM](#)

Advanced filtering options and fine-tuned controls, such as the ability to set time and size limitations, provide versatility. The capture can be stored in Azure Storage, on the VM's disk, or both. You can then analyze the capture file using several standard network capture analysis tools. Learn more about packet capture.

#### [Diagnose problems with an Azure Virtual network gateway and connections](#)

Virtual network gateways provide connectivity between on-premises resources and Azure virtual networks. Monitoring gateways and their connections are critical to ensuring communication is not broken. The VPN diagnostics capability provides the ability to diagnose gateways and connections. VPN diagnostics diagnoses the health of the gateway, or gateway connection, and informs you whether a gateway and gateway connections, are available. If the gateway or connection is not available, VPN diagnostics tells you why, so you can resolve the problem. Learn more about VPN diagnostics by completing the Diagnose a communication problem between networks tutorial.

#### [Determine relative latencies between Azure regions and internet service providers](#)

You can query Network Watcher for latency information between Azure regions and across internet service providers. When you know latencies between Azure regions and across Internet service providers, you can deploy Azure resources to optimize network response time. Learn more about relative latencies.

#### [View security rules for a network interface](#)

The effective security rules for a network interface are a combination of all security rules applied to the network interface, and the subnet the network interface is in. The security group view capability shows you all security rules applied to the

network interface, the subnet the network interface is in, and the aggregate of both. With an understanding of which rules are applied to a network interface, you can add, remove, or change rules, if they're allowing or denying traffic that you want to change. Learn more about security group view.

## Metrics

There are limits to the number of network resources that you can create within an Azure subscription and region. If you meet the limits, you're unable to create more resources within the subscription or region. The network subscription limit capability provides a summary of how many of each network resource you have deployed in a subscription and region, and what the limit is for the resource. The following picture shows the partial output for network resources deployed in the East US region for an example subscription:

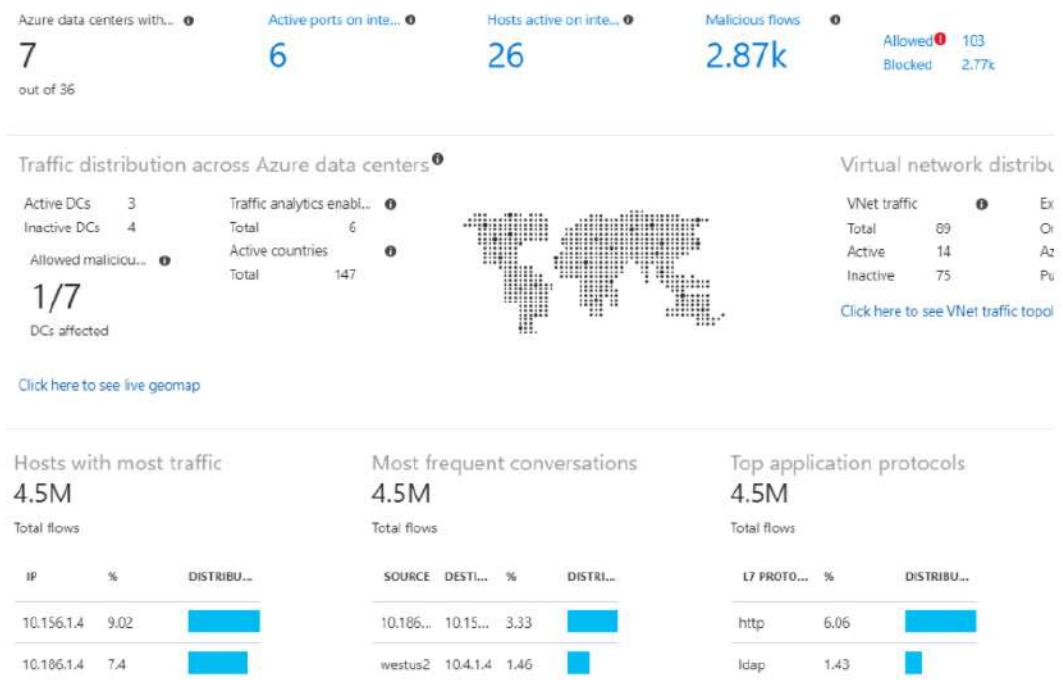
Subscription	Location	
	East US	
NAME	CURRENT LIMIT	USAGE
VirtualNetworks	50	2
StaticPublicIPAddresses	20	0
NetworkSecurityGroups	100	3
PublicIPAddresses	60	5
NetworkInterfaces	24000	4
LoadBalancers	100	0
ApplicationGateways	50	0
RouteTables	100	2

The information is helpful when planning future resource deployments.

## Logs

### Analyze traffic to or from a network security group

Network security groups (NSG) allow or deny inbound or outbound traffic to a network interface in a VM. The NSG flow log capability allows you to log the source and destination IP address, port, protocol, and whether traffic was allowed or denied by an NSG. You can analyze logs using a variety of tools, such as PowerBI and the traffic analytics capability. Traffic analytics provides rich visualizations of data written to NSG flow logs. The following picture shows some of the information and visualizations that traffic analytics presents from NSG flow log data:



### View diagnostic logs for network resources

You can enable diagnostic logging for Azure networking resources such as network security groups, public IP addresses, load balancers, virtual network gateways, and application gateways. The Diagnostic logs capability provides a single interface to enable and disable network resource diagnostic logs for any existing network resource that generates a diagnostic log. You can view diagnostic logs using tools such as Microsoft Power BI and Azure Monitor logs. To learn more about analyzing Azure network diagnostic logs, see Azure network solutions in Azure Monitor logs.

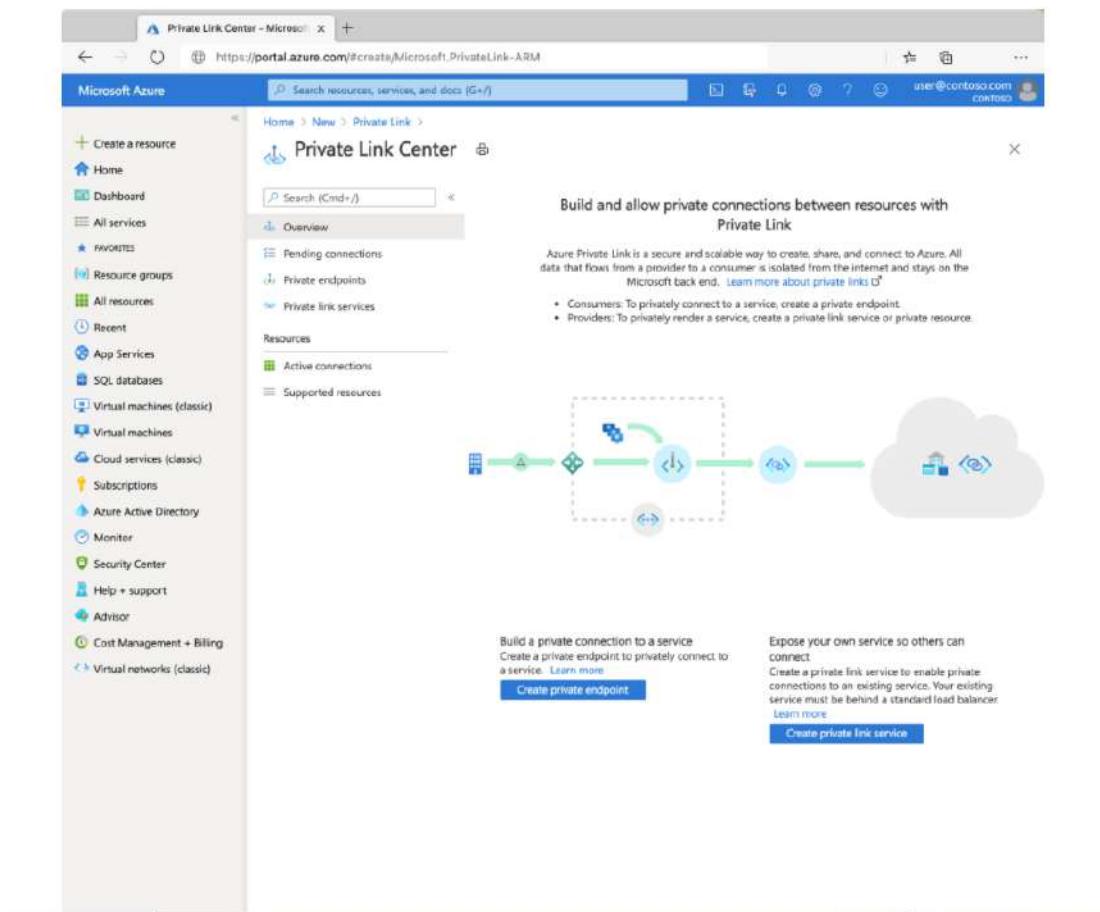
### Network Watcher automatic enablement

When you create or update a virtual network in your subscription, Network Watcher will be enabled automatically in your Virtual Network's region. There is no impact to your resources or associated charge for automatically enabling Network Watcher. For more information, see [Network Watcher create](#).

# Azure Private Link

Azure Private Link enables you to access Azure PaaS Services (for example, Azure Storage and SQL Database) and Azure hosted customer-owned/partner services over a private endpoint in your virtual network.

Traffic between your virtual network and the service travels the Microsoft backbone network. Exposing your service to the public internet is no longer necessary. You can create your own private link service in your virtual network and deliver it to your customers. Setup and consumption using Azure Private Link is consistent across Azure PaaS, customer-owned, and shared partner services.



## Key benefits

Azure Private Link provides the following benefits:

- Privately access services on the Azure platform: Connect your virtual network to services in Azure without a public IP address at the source or destination. Service providers can render their services in their own virtual network and consumers can access those services in their local virtual network. The Private

Link platform will handle the connectivity between the consumer and services over the Azure backbone network.

- On-premises and peered networks: Access services running in Azure from on-premises over ExpressRoute private peering, VPN tunnels, and peered virtual networks using private endpoints. There's no need to configure ExpressRoute Microsoft peering or traverse the internet to reach the service. Private Link provides a secure way to migrate workloads to Azure.
- Protection against data leakage: A private endpoint is mapped to an instance of a PaaS resource instead of the entire service. Consumers can only connect to the specific resource. Access to any other resource in the service is blocked. This mechanism provides protection against data leakage risks.
- Global reach: Connect privately to services running in other regions. The consumer's virtual network could be in region A and it can connect to services behind Private Link in region B.
- Extend to your own services: Enable the same experience and functionality to render your service privately to consumers in Azure. By placing your service behind a standard Azure Load Balancer, you can enable it for Private Link. The consumer can then connect directly to your service using a private endpoint in their own virtual network. You can manage the connection requests using an approval call flow. Azure Private Link works for consumers and services belonging to different Azure Active Directory tenants.

## Availability

Azure Private Link enables you to access Azure PaaS Services (for example, Azure Storage and SQL Database) and Azure hosted customer-owned/partner services over a private endpoint in your virtual network.

## Service availability

The following tables list the Private Link services and the regions where they're available.

AI + Machine Learning

AI + Machine Learning			
Supported services	Available regions	Other considerations	Status

Azure Machine Learning	All public regions		GA Learn how to create a private endpoint for Azure Machine Learning.
------------------------	--------------------	--	--

## Analytics

ANALYTICS			
Supported services	Available regions	Other considerations	Status
Azure Synapse Analytics	All public regions All Government regions	Supported for Proxy connection policy	GA Learn how to create a private endpoint for Azure Synapse Analytics.
Azure Event Hub	All public regions All Government regions		GA Learn how to create a private endpoint for Azure Event Hub.

ANALYTICS			
Supported services	Available regions	Other considerations	Status
Azure Monitor (Log Analytics & Application Insights)	All public regions		GA Learn how to create a private endpoint for Azure Monitor.
Azure Data Factory	All public regions All Government regions All China regions	Credentials need to be stored in an Azure key vault	GA Learn how to create a private endpoint for Azure Data Factory.
Azure HDInsight	All public regions All Government regions		GA Learn how to create a private endpoint for Azure HDInsight.

COMPUTE			
Supported services	Available regions	Other considerations	Status
Azure App Configuration	All public regions		GA Learn how to create a private endpoint for Azure App Configuration
Azure-managed Disks	All public regions All Government regions All China regions	Select for known limitations	GA Learn how to create a private endpoint for Azure Managed Disks.

## Containers

CONTAINERS			
Supported services	Available regions	Other considerations	Status
Azure Container Registry	All public regions All Government regions	Supported with premium tier of container registry. Select for tiers	GA Learn how to create a private endpoint for Azure Container Registry.

CONTAINERS			
Supported services	Available regions	Other considerations	Status
Azure Kubernetes Service - Kubernetes API	All public regions		GA Learn how to create a private endpoint for Azure Kubernetes Service.

#### Databases

DATABASES			
Supported services	Available regions	Other considerations	Status
Azure SQL Database	All public regions All Government regions All China regions	Supported for Proxy connection policy	GA Learn how to create a private endpoint for Azure SQL
Azure Cosmos DB	All public regions All Government regions All China regions		GA Learn how to create a private endpoint for Cosmos DB.

DATABASES			
Supported services	Available regions	Other considerations	Status
Azure Database for PostgreSQL - Single server	All public regions All Government regions All China regions	Supported for General Purpose and Memory Optimized pricing tiers	GA Learn how to create a private endpoint for Azure Database for PostgreSQL.
Azure Database for MySQL	All public regions All Government regions All China regions		GA Learn how to create a private endpoint for Azure Database for MySQL.
Azure Database for MariaDB	All public regions All Government regions All China regions		GA Learn how to create a private endpoint for Azure Database for MariaDB.

Integration

INTEGRATION			
Supported services	Available regions	Other considerations	Status
Azure Event Grid	All public regions All Government regions		GA Learn how to create a private endpoint for Azure Event Grid.
Azure Service Bus	All public region All Government regions	Supported with premium tier of Azure Service Bus. Select for tiers	GA Learn how to create a private endpoint for Azure Service Bus.

## Internet of Things (IoT)

INTERNET OF THINGS (IOT)			
Supported services	Available regions	Other considerations	Status
Azure IoT Hub	All public regions		GA Learn how to create a private endpoint for Azure IoT Hub.

INTERNET OF THINGS (IOT)			
Supported services	Available regions	Other considerations	Status
Azure Digital Twins	All public regions supported by Azure Digital Twins		Preview Learn how to create a private endpoint for Azure Digital Twins.

#### Management and Governance

MANAGEMENT AND GOVERNANCE			
Supported services	Available regions	Other considerations	Status
Azure Automation	All public regions All Government regions		GA Learn how to create a private endpoint for Azure Automation.
Azure Backup	All public regions All Government regions		GA Learn how to create a private endpoint for Azure Backup.

MANAGEMENT AND GOVERNANCE			
Supported services	Available regions	Other considerations	Status
Azure Purview	Southeast Asia, Australia East, Brazil South, North Europe, West Europe, Canada Central, East US, East US 2, EAST US 2 EUAP, South Central US, West Central US, West US 2, Central India, UK South	Select for known limitations	GA Learn how to create a private endpoint for Azure Purview.

## Security

SECURITY			
Supported services	Available regions	Other considerations	Status
Azure Key Vault	All public regions All Government regions		GA Learn how to create a private endpoint for Azure Key Vault.

## Storage

STORAGE			
Supported services	Available regions	Other considerations	Status
Azure Blob storage (including Data Lake Storage Gen2)	All public regions All Government regions	Supported only on Account Kind General Purpose V2	GA Learn how to create a private endpoint for blob storage.
Azure Files	All public regions All Government regions		GA Learn how to create Azure Files network endpoints.
Azure File Sync	All public regions		GA Learn how to create Azure Files network endpoints.
Azure Queue storage	All public regions All Government regions	Supported only on Account Kind General Purpose V2	GA Learn how to create a private endpoint for queue storage.

STORAGE			
Supported services	Available regions	Other considerations	Status
Azure Table storage	All public regions All Government regions	Supported only on Account Kind General Purpose V2	GA Learn how to create a private endpoint for table storage.
Azure Batch	All public regions except: Germany CENTRAL, Germany NORTHEAST All Government regions		GA Learn how to create a private endpoint for Azure Batch.

## Web

WEB			
Supported services	Available regions	Other considerations	Status
Azure SignalR	EAST US, SOUTH CENTRAL US, WEST US 2, All China regions		Preview Learn how to create a private endpoint for Azure SignalR.

WEB			
Supported services	Available regions	Other considerations	Status
Azure Web Apps	All public regions China North 2 & East 2	Supported with PremiumV2, PremiumV3, or Function Premium plan	GA Learn how to create a private endpoint for Azure Web Apps.
Azure Search	All public regions All Government regions	Supported with service in Private Mode	GA Learn how to create a private endpoint for Azure Search.
Azure Relay	All public regions		Preview Learn how to create a private endpoint for Azure Relay.

Private Link service with a standard load balancer

PRIVATE LINK SERVICE WITH A STANDARD LOAD BALANCER			
Supported services	Available regions	Other considerations	Status
Private Link services behind standard Azure Load Balancer	All public regions All Government regions All China regions	Supported on Standard Load Balancer	GA Learn how to create a private link service.

## Logging and Monitoring

Azure Private Link has integration with Azure Monitor. This combination allows:

- Archival of logs to a storage account.
- Streaming of events to your Event Hub.
- Azure Monitor logging.

You can access the following information on Azure Monitor:

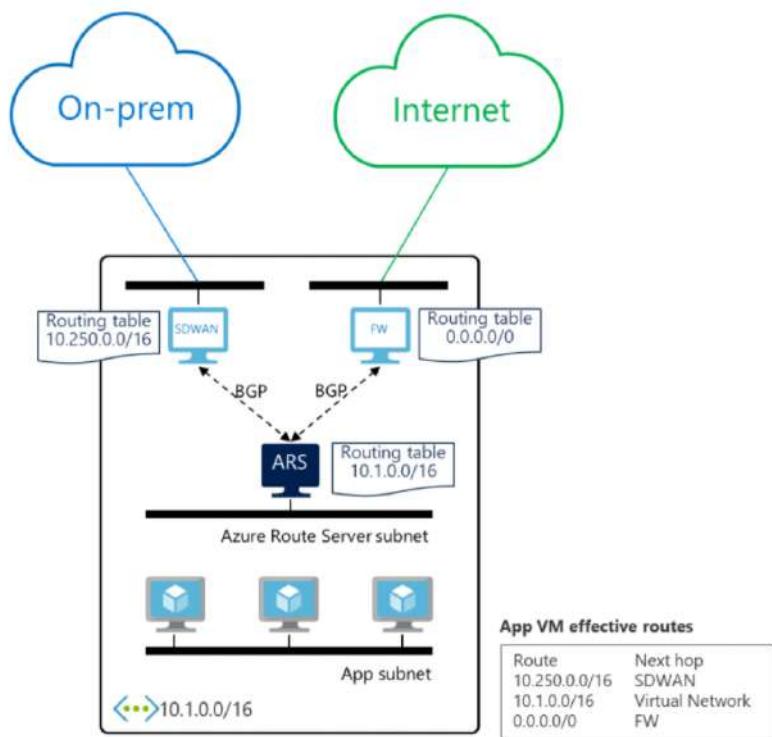
- Private endpoint:
  - Data processed by the Private Endpoint (IN/OUT)
- Private Link service:
  - Data processed by the Private Link service (IN/OUT)
  - NAT port availability

# Azure Route Server

Azure Route Server simplifies dynamic routing between your network virtual appliance (NVA) and your virtual network. It allows you to exchange routing information directly through Border Gateway Protocol (BGP) routing protocol between any NVA that supports the BGP routing protocol and the Azure Software Defined Network (SDN) in the Azure Virtual Network (VNET) without the need to manually configure or maintain route tables. Azure Route Server is a fully managed service and is configured with high availability.

## How does it work?

The following diagram illustrates how Azure Route Server works with an SDWAN NVA and a security NVA in a virtual network. Once you've established the BGP peering, Azure Route Server will receive an on-premises route (10.250.0.0/16) from the SDWAN appliance and a default route (0.0.0.0/0) from the firewall. These routes are then automatically configured on the VMs in the virtual network. As a result, all traffic destined to the on-premises network will be sent to the SDWAN appliance. While all Internet-bound traffic will be sent to the firewall. In the opposite direction, Azure Route Server will send the virtual network address (10.1.0.0/16) to both NVAs. The SDWAN appliance can propagate it further to the on-premises network.



## Key benefits

Azure Route Server simplifies configuration, management, and deployment of your NVA in your virtual network.

- You no longer need to manually update the routing table on your NVA whenever your virtual network addresses are updated.
- You no longer need to update User-Defined Routes manually whenever your NVA announces new routes or withdraw old ones.
- You can peer multiple instances of your NVA with Azure Route Server. You can configure the BGP attributes in your NVA and, depending on your design (e.g., active-active for performance or active-passive for resiliency), let Azure Route Server know which NVA instance is active or which one is passive.
- The interface between NVA and Azure Route Server is based on a common standard protocol. As long as your NVA supports BGP, you can peer it with Azure Route Server. For more information, see [Route Server supported routing protocols](#).
- You can deploy Azure Route Server in any of your new or existing virtual network.

## Azure Traffic Manager

Azure Traffic Manager is a DNS-based traffic load balancer. This service allows you to distribute traffic to your public facing applications across the global Azure regions. Traffic Manager also provides your public endpoints with high availability and quick responsiveness.

Traffic Manager uses DNS to direct the client requests to the appropriate service endpoint based on a traffic-routing method. Traffic manager also provides health monitoring for every endpoint. The endpoint can be any Internet-facing service hosted inside or outside of Azure. Traffic Manager provides a range of traffic-routing methods and endpoint monitoring options to suit different application needs and automatic failover models. Traffic Manager is resilient to failure, including the failure of an entire Azure region.

Traffic Manager offers the following features:

- Increase application availability

Traffic Manager delivers high availability for your critical applications by monitoring your endpoints and providing automatic failover when an endpoint goes down.

- Improve application performance

Azure allows you to run cloud services and websites in datacenters located around the world. Traffic Manager can improve the responsiveness of your website by directing traffic to the endpoint with the lowest latency.

- Service maintenance without downtime

You can have planned maintenance done on your applications without downtime. Traffic Manager can direct traffic to alternative endpoints while the maintenance is in progress.

- Combine hybrid applications

Traffic Manager supports external, non-Azure endpoints enabling it to be used with hybrid cloud and on-premises deployments, including the "burst-to-cloud," "migrate-to-cloud," and "failover-to-cloud" scenarios.

- Distribute traffic for complex deployments

Using nested Traffic Manager profiles, multiple traffic-routing methods can be combined to create sophisticated and flexible rules to scale to the needs of larger, more complex deployments.

## Azure Virtual Network

Azure Virtual Network (VNet) is the fundamental building block for your private network in Azure. VNet enables many types of Azure resources, such as Azure Virtual Machines (VM), to securely communicate with each other, the internet, and on-premises networks. VNet is similar to a traditional network that you'd operate in your own data center, but brings with it additional benefits of Azure's infrastructure such as scale, availability, and isolation.

### Why use an Azure Virtual network?

Azure virtual network enables Azure resources to securely communicate with each other, the internet, and on-premises networks. Key scenarios that you can accomplish with a virtual network include - communication of Azure resources with the internet, communication between Azure resources, communication with on-premises resources, filtering network traffic, routing network traffic, and integration with Azure services.

## Communicate with the internet

All resources in a VNet can communicate outbound to the internet, by default. You can communicate inbound to a resource by assigning a public IP address or a public Load Balancer. You can also use public IP or public Load Balancer to manage your outbound connections. To learn more about outbound connections in Azure, see [Outbound connections, Public IP addresses, and Load Balancer](#).

## Communicate between Azure resources

Azure resources communicate securely with each other in one of the following ways:

- Through a virtual network: You can deploy VMs, and several other types of Azure resources to a virtual network, such as Azure App Service Environments, the Azure Kubernetes Service (AKS), and Azure Virtual Machine Scale Sets. To view a complete list of Azure resources that you can deploy into a virtual network, see [Virtual network service integration](#).
- Through a virtual network service endpoint: Extend your virtual network private address space and the identity of your virtual network to Azure service resources, such as Azure Storage accounts and Azure SQL Database, over a direct connection. Service endpoints allow you to secure your critical Azure service resources to only a virtual network. To learn more, see [Virtual network service endpoints overview](#).
- Through VNet Peering: You can connect virtual networks to each other, enabling resources in either virtual network to communicate with each other, using virtual network peering. The virtual networks you connect can be in the same, or different, Azure regions. To learn more, see [Virtual network peering](#).

## Communicate with on-premises resources

You can connect your on-premises computers and networks to a virtual network using any combination of the following options:

- Point-to-site virtual private network (VPN): Established between a virtual network and a single computer in your network. Each computer that wants to establish connectivity with a virtual network must configure its connection. This connection type is great if you're just getting started with Azure, or for developers, because it requires little or no changes to your existing network. The communication between your computer and a virtual network is sent through an encrypted tunnel over the internet. To learn more, see [Point-to-site VPN](#).
- Site-to-site VPN: Established between your on-premises VPN device and an Azure VPN Gateway that is deployed in a virtual network. This connection type enables any on-premises resource that you authorize to access a virtual network. The communication between your on-premises VPN device and an Azure VPN

gateway is sent through an encrypted tunnel over the internet. To learn more, see Site-to-site VPN.

- Azure ExpressRoute: Established between your network and Azure, through an ExpressRoute partner. This connection is private. Traffic does not go over the internet. To learn more, see ExpressRoute.

## Filter network traffic

You can filter network traffic between subnets using either or both of the following options:

- Network security groups: Network security groups and application security groups can contain multiple inbound and outbound security rules that enable you to filter traffic to and from resources by source and destination IP address, port, and protocol. To learn more, see Network security groups or Application security groups.
- Network virtual appliances: A network virtual appliance is a VM that performs a network function, such as a firewall, WAN optimization, or other network function. To view a list of available network virtual appliances that you can deploy in a virtual network, see Azure Marketplace.

## Route network traffic

Azure routes traffic between subnets, connected virtual networks, on-premises networks, and the Internet, by default. You can implement either or both of the following options to override the default routes Azure creates:

- Route tables: You can create custom route tables with routes that control where traffic is routed to for each subnet. Learn more about route tables.
- Border gateway protocol (BGP) routes: If you connect your virtual network to your on-premises network using an Azure VPN Gateway or ExpressRoute connection, you can propagate your on-premises BGP routes to your virtual networks. Learn more about using BGP with Azure VPN Gateway and ExpressRoute.

## Virtual network integration for Azure services

Integrating Azure services to an Azure virtual network enables private access to the service from virtual machines or compute resources in the virtual network. You can integrate Azure services in your virtual network with the following options:

- Deploying dedicated instances of the service into a virtual network. The services can then be privately accessed within the virtual network and from on-premises networks.

- Using Private Link to access privately a specific instance of the service from your virtual network and from on-premises networks.
- You can also access the service using public endpoints by extending a virtual network to the service, through service endpoints. Service endpoints allow service resources to be secured to the virtual network.

## Azure VNet limits

There are certain limits around the number of Azure resources you can deploy. Most Azure networking limits are at the maximum values. However, you can increase certain networking limits as specified on the VNet limits page.

## Virtual networks and availability zones

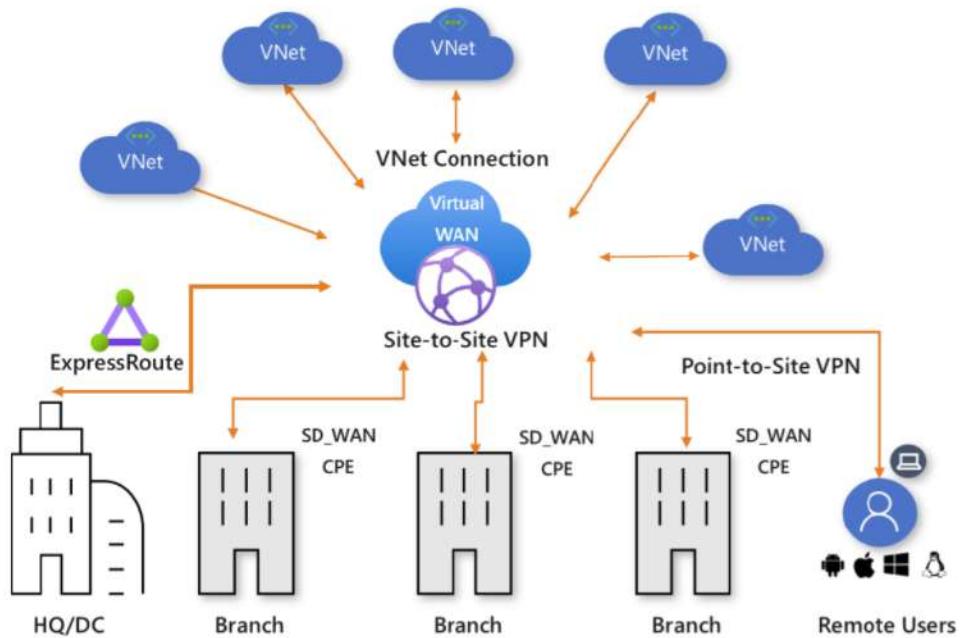
Virtual networks and subnets span all availability zones in a region. You don't need to divide them by availability zones to accommodate zonal resources. For example, if you configure a zonal VM, you don't have to take into consideration the virtual network when selecting the availability zone for the VM. The same is true for other zonal resources.

## Azure Virtual WAN

Azure Virtual WAN is a networking service that brings many networking, security, and routing functionalities together to provide a single operational interface. These functionalities include branch connectivity (via connectivity automation from Virtual WAN Partner devices such as SD-WAN or VPN CPE), Site-to-site VPN connectivity, remote user VPN (Point-to-site) connectivity, private (ExpressRoute) connectivity, intra-cloud connectivity (transitive connectivity for virtual networks), VPN ExpressRoute inter-connectivity, routing, Azure Firewall, and encryption for private connectivity. You do not have to have all of these use cases to start using Virtual WAN. You can simply get started with just one use case, and then adjust your network as it evolves.

The Virtual WAN architecture is a hub and spoke architecture with scale and performance built in for branches (VPN/SD-WAN devices), users (Azure VPN/OpenVPN/IKEv2 clients), ExpressRoute circuits, and virtual networks. It enables a global transit network architecture, where the cloud hosted network 'hub' enables transitive connectivity between endpoints that may be distributed across different types of 'spokes'.

Azure regions serve as hubs that you can choose to connect to. All hubs are connected in full mesh in a Standard Virtual WAN making it easy for the user to use the Microsoft backbone for any-to-any (any spoke) connectivity. For spoke connectivity with SD-WAN/VPN devices, users can either manually set it up in Azure Virtual WAN, or use the Virtual WAN CPE (SD-WAN/VPN) partner solution to set up connectivity to Azure. We have a list of partners that support connectivity automation (ability to export the device info into Azure, download the Azure configuration and establish connectivity) with Azure Virtual WAN. For more information, see the Virtual WAN partners and locations article.



This article provides a quick view into the network connectivity in Azure Virtual WAN. Virtual WAN offers the following advantages:

- Integrated connectivity solutions in hub and spoke: Automate site-to-site configuration and connectivity between on-premises sites and an Azure hub.
- Automated spoke setup and configuration: Connect your virtual networks and workloads to the Azure hub seamlessly.
- Intuitive troubleshooting: You can see the end-to-end flow within Azure, and then use this information to take required actions.

## Basic and Standard virtual WANs

There are two types of virtual WANs: Basic and Standard. The following table shows the available configurations for each type.

BASIC AND STANDARD VIRTUAL WANS		
Virtual WAN type	Hub type	Available configurations
Basic	Basic	Site-to-site VPN only
Standard	Standard	ExpressRoute User VPN (P2S) VPN (site-to-site) Inter-hub and VNet-to-VNet transiting through the virtual hub Azure Firewall NVA in a virtual WAN

## Architecture

For information about Virtual WAN architecture and how to migrate to Virtual WAN, see the following articles:

- [Virtual WAN architecture](#)
- [Global transit network architecture](#)

## Virtual WAN resources

To configure an end-to-end virtual WAN, you create the following resources:

- **VirtualWAN:** The virtualWAN resource represents a virtual overlay of your Azure network and is a collection of multiple resources. It contains links to all your virtual hubs that you would like to have within the virtual WAN. Virtual WAN resources are isolated from each other and cannot contain a common hub. Virtual hubs across Virtual WAN do not communicate with each other.
  - **Hub:** A virtual hub is a Microsoft-managed virtual network. The hub contains various service endpoints to enable connectivity. From your on-premises network (vpnsite), you can connect to a VPN Gateway inside the virtual hub, connect ExpressRoute circuits to a virtual hub, or even connect mobile users to a Point-to-site gateway in the virtual hub. The hub is the core of your network in a region. Multiple virtual hubs can be created in the same region.
- A hub gateway is not the same as a virtual network gateway that you use for ExpressRoute and VPN Gateway. For example, when using Virtual WAN, you don't create a site-to-site connection from your on-premises site directly to your VNet. Instead, you create a site-to-site connection to the hub. The traffic always goes through the hub gateway. This means that your VNets do not need their own

virtual network gateway. Virtual WAN lets your VNets take advantage of scaling easily through the virtual hub and the virtual hub gateway.

- Hub virtual network connection: The Hub virtual network connection resource is used to connect the hub seamlessly to your virtual network. One virtual network can be connected to only one virtual hub.
- Hub-to-Hub connection: Hubs are all connected to each other in a virtual WAN. This implies that a branch, user, or VNet connected to a local hub can communicate with another branch or VNet using the full mesh architecture of the connected hubs. You can also connect VNets within a hub transiting through the virtual hub, as well as VNets across hub, using the hub-to-hub connected framework.
- Hub route table: You can create a virtual hub route and apply the route to the virtual hub route table. You can apply multiple routes to the virtual hub route table.

## **Additional Virtual WAN resources**

Site: This resource is used for site-to-site connections only. The site resource is vpnsite. It represents your on-premises VPN device and its settings. By working with a Virtual WAN partner, you have a built-in solution to automatically export this information to Azure.

## **Connectivity**

### **Site-to-site VPN connections**

You can connect to your resources in Azure over a Site-to-site IPsec/IKE (IKEv2) connection. For more information, see [Create a site-to-site connection using Virtual WAN](#).

This type of connection requires a VPN device or a Virtual WAN Partner device. Virtual WAN partners provide automation for connectivity, which is the ability to export the device info into Azure, download the Azure configuration, and establish connectivity to the Azure Virtual WAN hub. For a list of the available partners and locations, see the [Virtual WAN partners and locations](#) article. If your VPN/SD-WAN device provider is not listed in the mentioned link, then you can simply use the step-by-step instruction [Create a site-to-site connection using Virtual WAN](#) to set up the connection.

## User VPN (point-to-site) connections

You can connect to your resources in Azure over an IPsec/IKE (IKEv2) or OpenVPN connection. This type of connection requires a VPN client to be configured on the client computer. For more information, see [Create a point-to-site connection](#).

## ExpressRoute connections

ExpressRoute lets you connect on-premises network to Azure over a private connection. To create the connection, see [Create an ExpressRoute connection using Virtual WAN](#).

## Hub-to-VNet connections

You can connect an Azure virtual network to a virtual hub. For more information, see [Connect your VNet to a hub](#).

## Transit connectivity

### ➤ Transit connectivity between VNets

Virtual WAN allows transit connectivity between VNets. VNets connect to a virtual hub via a virtual network connection. Transit connectivity between the VNets in Standard Virtual WAN is enabled due to the presence of a router in every virtual hub. This router is instantiated when the virtual hub is first created.

The router can have four routing statuses: Provisioned, Provisioning, Failed, or None. The Routing status is located in the Azure portal by navigating to the Virtual Hub page.

- A None status indicates that the Virtual hub did not provision the router. This can happen if the Virtual WAN is of type Basic, or if the virtual hub was deployed prior to the service being made available.
- A Failed status indicates failure during instantiation. In order to instantiate or reset the router, you can locate the Reset Router option by navigating to the virtual hub Overview page in the Azure portal.

Every virtual hub router supports an aggregate throughput up to 50 Gbps.

Connectivity between the virtual network connections assumes, by default, a maximum total of 2000 VM workload across all VNets connected to a single virtual Hub. This limit can be increased opening an online customer support request. For cost implication, see [Routing Infrastructure Unit cost](#) in the [Azure Virtual WAN Pricing page](#).

### ➤ Transit connectivity between VPN and ExpressRoute

Virtual WAN allows transit connectivity between VPN and ExpressRoute. This implies that VPN-connected sites or remote users can communicate with ExpressRoute-connected sites. There is also an implicit assumption that the Branch-

to-branch flag is enabled and BGP is supported in VPN and ExpressRoute connections. This flag can be located in the Azure Virtual WAN settings in Azure portal. All route management is provided by the virtual hub router, which also enables transit connectivity between virtual networks.

## Custom Routing

Virtual WAN provides advanced routing enhancements. Ability to set up custom route tables, optimize virtual network routing with route association and propagation, logically group route tables with labels and simplify numerous network virtual appliance (NVA) or shared services routing scenarios.

## Global VNet peering

Global VNet Peering provides a mechanism to connect two VNets in different regions. In Virtual WAN, virtual network connections connect VNets to virtual hubs. The user does not need to set up global VNet peering explicitly. VNets connected to virtual hub in same region incur VNet peering charges. VNets connected to virtual hub in a different region incur Global VNet peering charges.

## ExpressRoute traffic encryption

Azure Virtual WAN provides ability to encrypt your ExpressRoute traffic. The technique provides an encrypted transit between the on-premises networks and Azure virtual networks over ExpressRoute, without going over the public internet or using public IP addresses. For more information, see IPsec over ExpressRoute for Virtual WAN.

## Locations

### Route tables for Basic and Standard virtual WANs

Route tables now have features for association and propagation. A pre-existing route table is a route table that does not have these features. If you have pre-existing routes in hub routing and would like to use the new capabilities, consider the following:

Standard Virtual WAN Customers with pre-existing routes in virtual hub: If you have pre-existing routes in the Routing section for the hub in the Azure portal, you will need to first delete them and then attempt creating new route tables (available in the Route Tables section for the hub in Azure portal). It is highly encouraged to do the delete step for all hubs in a Virtual WAN.

Basic Virtual WAN Customers with pre-existing routes in virtual hub: If you have pre-existing routes in Routing section for the hub in the Azure portal, you will

need to first delete them, then upgrade your Basic Virtual WAN to Standard Virtual WAN. See [Upgrade a virtual WAN from Basic to Standard](#). It is highly encouraged to do the delete step for all hubs in a Virtual WAN.

### Gated public preview

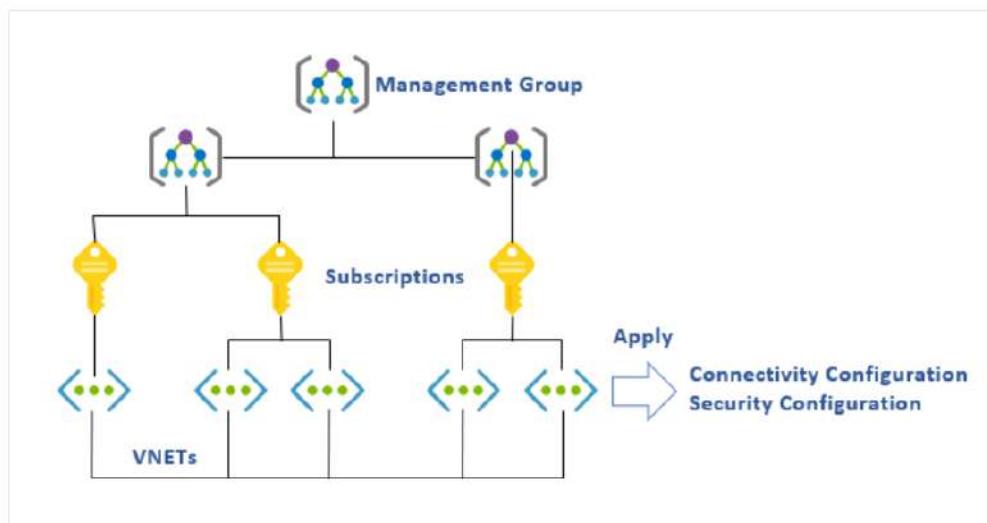
The below features are currently in gated public preview.

GATED PUBLIC PREVIEW	
Feature	Description
Routing Intent and Policies Enabling Inter-hub security	This feature allows customers to configure internet-bound, private or inter-hub traffic flow through the Azure Firewall. Please review <a href="#">Routing Intent and Policies</a> to learn more.
Hub to Hub over ER preview link	This feature allows traffic between 2 hubs traverse through the Azure Virtual WAN router in each hub and uses a hub-to-hub path instead of the ExpressRoute path (which traverses through the Microsoft edge routers/MSEE). Please review <a href="#">Hub to Hub over ER preview link</a>
BGP peering with a virtual hub	This feature provides the ability for the virtual hub to pair with and directly exchange routing information through Border Gateway Protocol (BGP) routing protocol. Please review the concept <a href="#">BGP peering with a virtual hub</a> and the guide <a href="#">How to peer BGP with a virtual hub</a>

# Azure Virtual Network Manager

Azure Virtual Network Manager is a management service that enables you to group, configure, deploy, and manage virtual networks globally across subscriptions. With Virtual Network Manager, you can define network groups to identify and logically segment your virtual networks. Then you can determine the connectivity and security configurations you want and apply them across all the selected virtual networks in network groups at once.

## How does Azure Virtual Network Manager work?



During the creation process, you define the scope for what your Azure Virtual Network Manager will manage. Defining a scope requires a management group to be created. After defining the scope, you enable features such as Connectivity and the SecurityAdmin role for your Virtual Network Manager.

After you deploy the Virtual Network Manager instance, you then create a network group by using conditional statements to select virtual networks by name, tags, or IDs (dynamic membership). You can also select specific virtual networks (static membership). The network group rules defined are reflected in Azure Policy as a custom initiative definition and corresponding assignment that illustrate the rules you defined for virtual network membership. For more information about Azure Policy initiatives, see Azure Policy initiative structure. These policies are available in read-only mode today. For more information about how to create, update, and delete these policies, see Network groups and Azure Policy. You then create connectivity and/or security configuration(s) applied to those network groups based on your topology and security needs.

A connectivity configuration enables you to create a mesh or a hub-and-spoke network topology. A security configuration allows you to define a collection of rules that you can apply to one or more network groups at the global level. Once you've created your desired network groups and configurations, you can deploy the configurations to any region of your choosing.

## Key benefits

- Centrally manage connectivity and security policies globally across regions and subscriptions.
- Enable transitive communication between spokes in a hub-and-spoke configuration without the complexity of managing a mesh network.
- Highly scalable and highly available service with redundancy and replication across the globe.
- Ability to create global network security rules that override network security group rules.
- Low latency and high bandwidth between resources in different virtual networks using virtual network peering.
- Roll out network changes through a specific region sequence and frequency of your choosing.

## Public preview regions

- North Central US
- West US
- West US 2
- East US
- East US 2
- North Europe
- West Europe
- France Central

# Azure VPN Gateway

A VPN gateway is a specific type of virtual network gateway that is used to send encrypted traffic between an Azure virtual network and an on-premises location over the public Internet. You can also use a VPN gateway to send encrypted traffic between Azure virtual networks over the Microsoft network. Each virtual network can have only one VPN gateway. However, you can create multiple connections to the same VPN gateway. When you create multiple connections to the same VPN gateway, all VPN tunnels share the available gateway bandwidth.

## What is a virtual network gateway?

A virtual network gateway is composed of two or more VMs that are deployed to a specific subnet you create called the gateway subnet. Virtual network gateway VMs contain routing tables and run specific gateway services. These VMs are created when you create the virtual network gateway. You can't directly configure the VMs that are part of the virtual network gateway.

When you configure a virtual network gateway, you configure a setting that specifies the gateway type. The gateway type determines how the virtual network gateway will be used and the actions that the gateway takes. The gateway type 'Vpn' specifies that the type of virtual network gateway created is a 'VPN gateway'. This distinguishes it from an ExpressRoute gateway, which uses a different gateway type. A virtual network can have two virtual network gateways; one VPN gateway and one ExpressRoute gateway. For more information, see [Gateway types](#).

Creating a gateway can often take 45 minutes or more, depending on the selected gateway SKU. When you create a virtual network gateway, gateway VMs are deployed to the gateway subnet and configured with the settings that you specify. After you create a VPN gateway, you can create an IPsec/IKE VPN tunnel connection between that VPN gateway and another VPN gateway (VNet-to-VNet), or create a cross-premises IPsec/IKE VPN tunnel connection between the VPN gateway and an on-premises VPN device (Site-to-Site). You can also create a Point-to-Site VPN connection (VPN over OpenVPN, IKEv2, or SSTP), which lets you connect to your virtual network from a remote location, such as from a conference or from home.

## Configuring a VPN Gateway

A VPN gateway connection relies on multiple resources that are configured with specific settings. Most of the resources can be configured separately, although some resources must be configured in a certain order.

## Design

It's important to know that there are different configurations available for VPN gateway connections. You need to determine which configuration best fits your needs. For example, Point-to-Site, Site-to-Site, and coexisting ExpressRoute/Site-to-Site connections all have different instructions and configuration requirements. For information about design and to view connection topology diagrams, see [Design](#).

## Planning table

The following table can help you decide the best connectivity option for your solution.

PLANNING TABLE			
	Point-to-Site	Site-to-Site	ExpressRoute
Azure Supported Services	Cloud Services and Virtual Machines	Cloud Services and Virtual Machines	Services list
Typical Bandwidths	Based on the gateway SKU	Typically < 1 Gbps aggregate	50 Mbps, 100 Mbps, 200 Mbps, 500 Mbps, 1 Gbps, 2 Gbps, 5 Gbps, 10 Gbps
Protocols Supported	Secure Sockets Tunneling Protocol (SSTP), OpenVPN and IPsec	IPsec	Direct connection over VLANs, NSP's VPN technologies (MPLS, VPLS,...)
Routing	RouteBased (dynamic)	We support PolicyBased (static routing) and RouteBased (dynamic routing VPN)	BGP

PLANNING TABLE			
	Point-to-Site	Site-to-Site	ExpressRoute
Connection resiliency	active-passive	active-passive or active-active	active-active
Typical use case	Secure access to Azure virtual networks for remote users	Dev / test / lab scenarios and small to medium scale production workloads for cloud services and virtual machines	Access to all Azure services (validated list), Enterprise-class and mission critical workloads, Backup, Big Data, Azure as a DR site
SLA	SLA	SLA	SLA
Pricing	Pricing	Pricing	Pricing
Technical Documentation	VPN Gateway Documentation	VPN Gateway Documentation	ExpressRoute Documentation
FAQ	VPN Gateway FAQ	VPN Gateway FAQ	ExpressRoute FAQ

## Settings

The settings that you chose for each resource are critical to creating a successful connection. For information about individual resources and settings for VPN Gateway, see [About VPN Gateway settings](#). The article contains information to help you understand gateway types, gateway SKUs, VPN types, connection types, gateway subnets, local network gateways, and various other resource settings that you may want to consider.

## Deployment tools

You can start out creating and configuring resources using one configuration tool, such as the Azure portal. You can later decide to switch to another tool, such as PowerShell, to configure additional resources, or modify existing resources when applicable. Currently, you can't configure every resource and resource setting in the Azure portal. The instructions in the articles for each connection topology specify when a specific configuration tool is needed.

## Gateway SKUs

When you create a virtual network gateway, you specify the gateway SKU that you want to use. Select the SKU that satisfies your requirements based on the types of workloads, throughputs, features, and SLAs.

- For more information about gateway SKUs, including supported features, production and dev-test, and configuration steps, see the [VPN Gateway Settings - Gateway SKUs](#) article.
- For Legacy SKU information, see [Working with Legacy SKUs](#).

GATEWAY SKUS BY TUNNEL, CONNECTION, AND PORT							
VPN Gateway Generation	SKU	S2S/VNet-to-VNet Tunnels	P2S SSTP Connections	P2S IKEv2/OpenVPN Connections	Aggregate Throughput Benchmark	BGP	Zone-redundant
Generation1	Basic	Max. 10	Max. 128	Not Supported	100 Mbps	Not Supported	No
Generation1	VpnGw1	Max. 30	Max. 128	Max. 250	650 Mbps	Supported	No
Generation1	VpnGw2	Max. 30	Max. 128	Max. 500	1 Gbps	Supported	No
Generation1	VpnGw3	Max. 30	Max. 128	Max. 1000	1.25 Gbps	Supported	No
Generation1	VpnGw1AZ	Max. 30	Max. 128	Max. 250	650 Mbps	Supported	Yes
Generation1	VpnGw2AZ	Max. 30	Max. 128	Max. 500	1 Gbps	Supported	Yes

GATEWAY SKUS BY TUNNEL, CONNECTION, AND								
VPN Generation	SKU	S2S/VNet-to-VNet Tunne ls	P2S SSTP Connections	P2S IKEv2/OpenVPN Connections	Aggregate Throughput Benchmark	BGP	Zone-redundant	
Generation1	VpnGw3AZ	Max. 30	Max. 128	Max. 1000	1.25 Gbps	Supported	Yes	
Generation2	VpnGw2	Max. 30	Max. 128	Max. 500	1.25 Gbps	Supported	No	
Generation2	VpnGw3	Max. 30	Max. 128	Max. 1000	2.5 Gbps	Supported	No	
Generation2	VpnGw4	Max. 100*	Max. 128	Max. 5000	5 Gbps	Supported	No	
Generation2	VpnGw5	Max. 100*	Max. 128	Max. 10000	10 Gbps	Supported	No	
Generation2	VpnGw2AZ	Max. 30	Max. 128	Max. 500	1.25 Gbps	Supported	Yes	
Generation2	VpnGw3AZ	Max. 30	Max. 128	Max. 1000	2.5 Gbps	Supported	Yes	
Generation2	VpnGw4AZ	Max. 100*	Max. 128	Max. 5000	5 Gbps	Supported	Yes	
Generation2	VpnGw5AZ	Max. 100*	Max. 128	Max. 10000	10 Gbps	Supported	Yes	

## Gateway SKUs by tunnel, connection, and throughput

(\*) Use Virtual WAN if you need more than 100 S2S VPN tunnels.

- The resizing of VpnGw SKUs is allowed within the same generation, except resizing of the Basic SKU. The Basic SKU is a legacy SKU and has feature limitations. In order to move from Basic to another VpnGw SKU, you must delete the Basic SKU VPN gateway and create a new gateway with the desired Generation and SKU size combination. You can only resize a Basic gateway to another legacy SKU (see Working with Legacy SKUs).
- These connection limits are separate. For example, you can have 128 SSTP connections and also 250 IKEv2 connections on a VpnGw1 SKU.

- Pricing information can be found on the Pricing page.
- SLA (Service Level Agreement) information can be found on the SLA page.
- On a single tunnel a maximum of 1 Gbps throughput can be achieved. Aggregate Throughput Benchmark in the above table is based on measurements of multiple tunnels aggregated through a single gateway. The Aggregate Throughput Benchmark for a VPN Gateway is S2S + P2S combined. If you have a lot of P2S connections, it can negatively impact a S2S connection due to throughput limitations. The Aggregate Throughput Benchmark is not a guaranteed throughput due to Internet traffic conditions and your application behaviors.
- To help our customers understand the relative performance of SKUs using different algorithms, we used publicly available iPerf and CTSTraffic tools to measure performances. The table below lists the results of performance tests for Generation 1, VpnGw SKUs. As you can see, the best performance is obtained when we used GCMAES256 algorithm for both IPsec Encryption and Integrity. We got average performance when using AES256 for IPsec Encryption and SHA256 for Integrity. When we used DES3 for IPsec Encryption and SHA256 for Integrity we got lowest performance.
- A VPN tunnel connects to a VPN gateway instance. Each instance throughput is mentioned in the above throughput table and is available aggregated across all tunnels connecting to that instance.

**TABLE 3**

Generation	SKU	Algorithms used	Throughput observed per tunnel	Packets per second per tunnel observed
Generation1	VpnGw1	GCMAES256 AES256 & SHA256 DES3 & SHA256	650 Mbps 500 Mbps 120 Mbps	58,000 50,000 50,000
Generation1	VpnGw2	GCMAES256 AES256 & SHA256 DES3 & SHA256	1 Gbps 500 Mbps 120 Mbps	90,000 80,000 55,000

TABLE 3

Generation	SKU	Algorithms used	Throughput observed per tunnel	Packets per second per tunnel observed
Generation1	VpnGw3	GCMAES256 AES256 & SHA256 DES3 & SHA256	1.25 Gbps 550 Mbps 120 Mbps	105,000 90,000 60,000
Generation1	VpnGw1 AZ	GCMAES256 AES256 & SHA256 DES3 & SHA256	650 Mbps 500 Mbps 120 Mbps	58,000 50,000 50,000
Generation1	VpnGw2 AZ	GCMAES256 AES256 & SHA256 DES3 & SHA256	1 Gbps 500 Mbps 120 Mbps	90,000 80,000 55,000
Generation1	VpnGw3 AZ	GCMAES256 AES256 & SHA256 DES3 & SHA256	1.25 Gbps 550 Mbps 120 Mbps	105,000 90,000 60,000

## Availability Zones

VPN gateways can be deployed in Azure Availability Zones. This brings resiliency, scalability, and higher availability to virtual network gateways. Deploying gateways in Azure Availability Zones physically and logically separates gateways within a region, while protecting your on-premises network connectivity to Azure from zone-level failures. see About zone-redundant virtual network gateways in Azure Availability Zones.

# AZURE STORAGE

## 1. Microsoft Azure Storage Overview

The Azure Storage platform is Microsoft's cloud storage solution for modern data storage scenarios. Core storage services offer a massively scalable object store for data objects, disk storage for Azure virtual machines (VMs), a file system service for the cloud, a messaging store for reliable messaging, and a NoSQL store. The services are:

- **Durable and highly available.** Redundancy ensures that your data is safe in the event of transient hardware failures. You can also opt to replicate data across datacentres or geographical regions for additional protection from local catastrophe or natural disaster. Data replicated in this way remains highly available in the event of an unexpected outage.
- **Secure.** All data written to an Azure storage account is encrypted by the service. Azure Storage provides you with fine-grained control over who has access to your data.
- **Scalable.** Azure Storage is designed to be massively scalable to meet the data storage and performance needs of today's applications.
- **Managed.** Azure handles hardware maintenance, updates, and critical issues for you.
- **Accessible.** Data in Azure Storage is accessible from anywhere in the world over HTTP or HTTPS. Microsoft provides client libraries for Azure Storage in a variety of languages, including .NET, Java, Node.js, Python, PHP, Ruby, Go, and others, as well as a mature REST API. Azure Storage supports scripting in Azure PowerShell or Azure CLI. And the Azure portal and Azure Storage Explorer offer easy visual solutions for working with your data.

## Core Azure Storage Services

The Azure Storage platform includes the following data services:

- Azure Blobs: A massively scalable object store for text and binary data. Also includes support for big data analytics through Data Lake Storage Gen2.

- [Azure Files](#): Managed file shares for cloud or on-premises deployments.
- [Azure Queues](#): A messaging store for reliable messaging between application components.
- [Azure Tables](#): A NoSQL store for schemaless storage of structured data.
- [Azure Disks](#): Block-level storage volumes for Azure VMs.

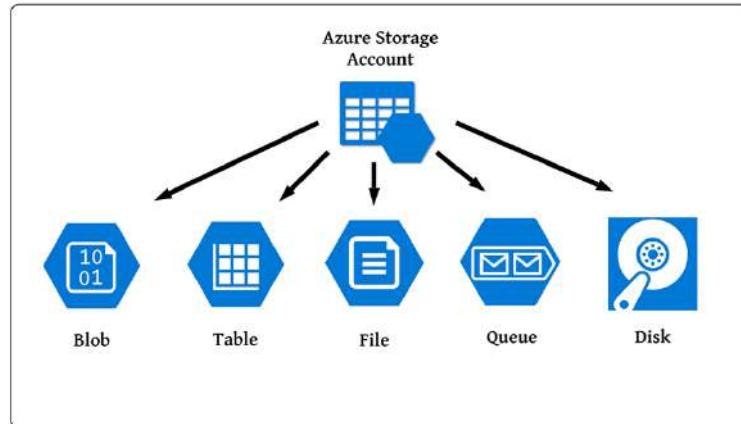


Fig: Azure Storage Services

## 1. Azure Blob Storage



Azure Blob storage is Microsoft's object storage solution for the cloud. Blob storage is optimized for storing massive amounts of unstructured data, such as text or binary data.

Blob storage is ideal for:

- Serving images or documents directly to a browser.
- Storing files for distributed access.
- Streaming video and audio.
- Storing data for backup and restore, disaster recovery, and archiving.
- Storing data for analysis by an on-premises or Azure-hosted service.

Users or client applications can access objects in Blob storage via HTTP/HTTPS, from anywhere in the world. Objects in Blob storage are accessible via the [Azure Storage REST API](#), [Azure PowerShell](#), [Azure CLI](#), or an Azure Storage client library. Client libraries are available for different languages, including:

- [.NET](#)
- [Java](#)
- [Node.js](#)
- [Python](#)
- [Go](#)
- [PHP](#)
- [Ruby](#)

### 1.1.1 Azure Data Lake Storage Gen2

Blob storage supports Azure Data Lake Storage Gen2, Microsoft's enterprise big data analytics solution for the cloud. Azure Data Lake Storage Gen2 offers a hierarchical file system as well as the advantages of Blob storage, including:

- Low-cost, tiered storage
- High availability
- Strong consistency
- Disaster recovery capabilities

### 1.1.2 Blob storage resources

Blob storage offers three types of resources:

- The **storage account**
- A **container** in the storage account
- A **blob** in a container

The following diagram shows the relationship between these resources.

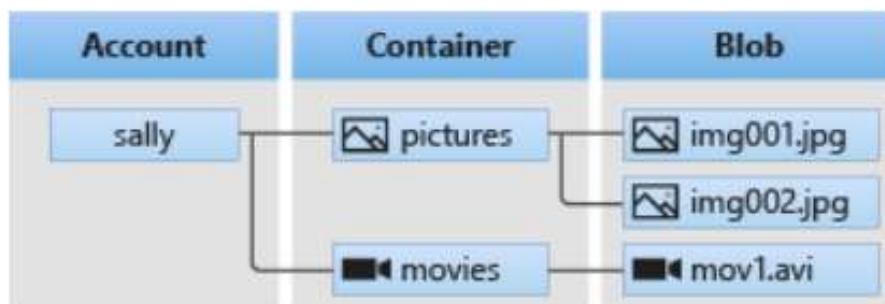


Fig: Relationship between the resources

### 1.1.2.1 Storage accounts

A storage account provides a unique namespace in Azure for your data. Every object that you store in Azure Storage has an address that includes your unique account name. The combination of the account name and the Azure Storage blob endpoint forms the base address for the objects in your storage account.

For example, if your storage account is named *myaccount*, then the default endpoint for Blob storage is: “ <http://myaccount.blob.core.windows.net> ”

### 1.1.2.2 Containers

A container organizes a set of blobs, similar to a directory in a file system. A storage account can include an unlimited number of containers, and a container can store an unlimited number of blobs.

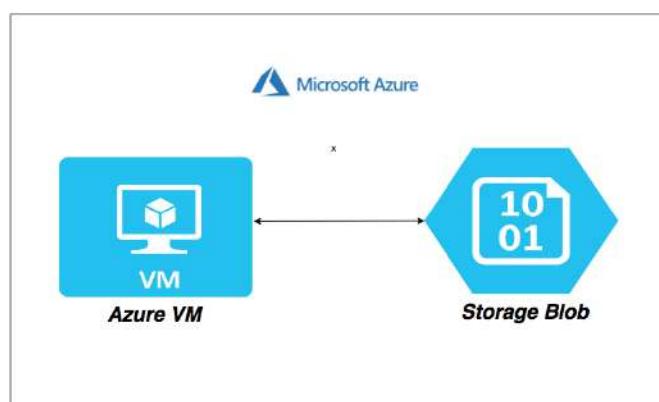
- The container name must be lowercase. For more information about naming containers, see [Naming and Referencing Containers, Blobs, and Metadata](#).

### 1.1.2.3 Blobs

Azure Storage supports three types of blobs:

- **Block blobs** store text and binary data. Block blobs are made up of blocks of data that can be managed individually. Block blobs can store up to about 190.7 TiB.
- **Append blobs** are made up of blocks like block blobs, but are optimized for append operations. Append blobs are ideal for scenarios such as logging data from virtual machines.
- **Page blobs** store random access files up to 8 TiB in size. Page blobs store virtual hard drive (VHD) files and serve as disks for Azure virtual machines.

### 1.1.3 Move data to Blob storage



A number of solutions exist for migrating existing data to Blob storage:

- **AzCopy** is an easy-to-use command-line tool for Windows and Linux that copies data to and from Blob storage, across containers, or across storage accounts.
- The **Azure Storage Data Movement library** is a .NET library for moving data between Azure Storage services. The AzCopy utility is built with the Data Movement library
- **Azure Data Factory** supports copying data to and from Blob storage by using the account key, a shared access signature, a service principal, or managed identities for Azure resources.
- **Blobfuse** is a virtual file system driver for Azure Blob storage. You can use blobfuse to access your existing block blob data in your Storage account through the Linux file system.
- **Azure Data Box** service is available to transfer on-premises data to Blob storage when large datasets or network constraints make uploading data over the wire unrealistic. Depending on your data size, you can request [Azure Data Box Disk](#), [Azure Data Box](#), or [Azure Data Box Heavy](#) devices from Microsoft. You can then copy your data to those devices and ship them back to Microsoft to be uploaded into Blob storage.
- The **Azure Import/Export service** provides a way to import or export large amounts of data to and from your storage account using hard drives that you provide.

## 1.2 Azure Files

Azure Files offers fully managed file shares in the cloud that are accessible via the industry standard [Server Message Block \(SMB\) protocol](#) or [Network File System \(NFS\) protocol](#). Azure Files file shares can be mounted concurrently by cloud or on-premises deployments. SMB Azure file shares are accessible from Windows, Linux, and macOS clients. NFS Azure Files shares are accessible from Linux or macOS clients. Additionally, SMB Azure file shares can be cached on Windows Servers with [Azure File Sync](#) for fast access near where the data is being used.

### 1.2.1 Why Azure Files is useful

- **Replace or supplement on-premises file servers:**

Azure Files can be used to completely replace or supplement traditional on-premises file servers or NAS devices. Popular operating systems such as Windows, macOS, and Linux can directly mount Azure file shares wherever they are in the world. SMB Azure file shares can also be replicated with Azure

File Sync to Windows Servers, either on-premises or in the cloud, for performance and distributed caching of the data where it's being used. With the recent release of [Azure Files AD Authentication](#), SMB Azure file shares can continue to work with AD hosted on-premises for access control.

- **"Lift and shift" applications:**

Azure Files makes it easy to "lift and shift" applications to the cloud that expect a file share to store file application or user data. Azure Files enables both the "classic" lift and shift scenario, where both the application and its data are moved to Azure, and the "hybrid" lift and shift scenario, where the application data is moved to Azure Files, and the application continues to run on-premises.

- **Simplify cloud development:**

Azure Files can also be used in numerous ways to simplify new cloud development projects. For example:

- **Shared application settings:**

A common pattern for distributed applications is to have configuration files in a centralized location where they can be accessed from many application instances. Application instances can load their configuration through the File REST API, and humans can access them as needed by mounting the SMB share locally.

- **Diagnostic share:**

An Azure file share is a convenient place for cloud applications to write their logs, metrics, and crash dumps. Logs can be written by the application instances via the File REST API, and developers can access them by mounting the file share on their local machine. This enables great flexibility, as developers can embrace cloud development without having to abandon any existing tooling they know and love.

- **Dev/Test/Debug:**

When developers or administrators are working on VMs in the cloud, they often need a set of tools or utilities. Copying such utilities and tools to each VM can be a time consuming exercise. By mounting an Azure file share locally on the VMs, a developer and administrator can quickly access their tools and utilities, no copying required.

- **Containerization:**

Azure file shares can be used as persistent volumes for stateful containers. Containers deliver "build once, run anywhere" capabilities that enable developers to accelerate innovation. For the containers that access raw data at every start, a shared file system is required to allow these containers to access the file system no matter which instance they run on.

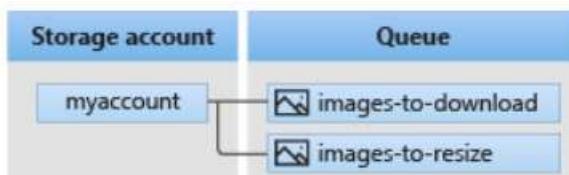
## 1.2.2 Key benefits

- **Shared access.** Azure file shares support the industry standard SMB and NFS protocols, meaning you can seamlessly replace your on-premises file shares with Azure file shares without worrying about application compatibility. Being able to share a file system across multiple machines, applications/instances is a significant advantage with Azure Files for applications that need shareability.
- **Fully managed.** Azure file shares can be created without the need to manage hardware or an OS. This means you don't have to deal with patching the server OS with critical security upgrades or replacing faulty hard disks.
- **Scripting and tooling.** PowerShell cmdlets and Azure CLI can be used to create, mount, and manage Azure file shares as part of the administration of Azure applications. You can create and manage Azure file shares using Azure portal and Azure Storage Explorer.
- **Resiliency.** Azure Files has been built from the ground up to be always available. Replacing on-premises file shares with Azure Files means you no longer have to wake up to deal with local power outages or network issues.
- **Familiar programmability.** Applications running in Azure can access data in the share via file [system I/O APIs](#). Developers can therefore leverage their existing code and skills to migrate existing applications. In addition to System IO APIs, you can use [Azure Storage Client Libraries](#) or the [Azure Storage REST API](#).

## 1.3 Queue storage

Azure Queue Storage is a service for storing large numbers of messages. You access messages from anywhere in the world via authenticated calls using HTTP or HTTPS. A queue message can be up to 64 KB in size. A queue may contain millions of messages, up to the total capacity limit of a storage account. Queues are commonly used to create a backlog of work to process asynchronously.

### 1.3.1 Queue Storage concepts:



- **URL format:** Queues are addressable using the following URL format:

`https://<storage account>.queue.core.windows.net/<queue>`

The following URL addresses a queue in the diagram:

<https://myaccount.queue.core.windows.net/images-to-download>

- **Storage account:** All access to Azure Storage is done through a storage account.
- **Queue:** A queue contains a set of messages. The queue name **must** be all lowercase.
- **Message:** A message, in any format, of up to 64 KB. Before version 2017-07-29, the maximum time-to-live allowed is seven days. For version 2017-07-29 or later, the maximum time-to-live can be any positive number, or -1 indicating that the message doesn't expire. If this parameter is omitted, the default time-to-live is seven days.

## 1.4 Azure Table Storage

Azure Table storage is a service that stores non-relational structured data (also known as structured NoSQL data) in the cloud, providing a key/attribute store with a schemaless design. Because Table storage is schemaless, it's easy to adapt your data as the needs of your application evolve. Access to Table storage data is fast and cost-effective for many types of applications, and is typically lower in cost than traditional SQL for similar volumes of data.

You can use Table storage to store flexible datasets like user data for web applications, address books, device information, or other types of metadata your service requires. You can store any number of entities in a table, and a storage account may contain any number of tables, up to the capacity limit of the storage account.

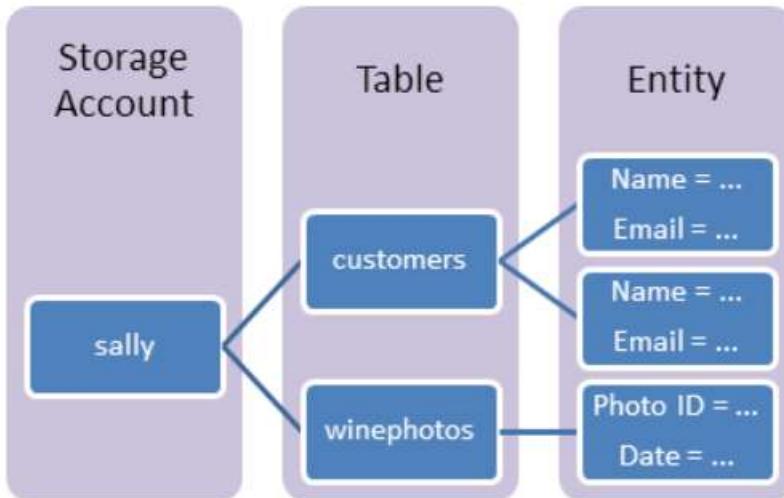
### 1.4.1 What is Table storage

Azure Table storage stores large amounts of structured data. The service is a NoSQL datastore which accepts authenticated calls from inside and outside the Azure cloud. Azure tables are ideal for storing structured, non-relational data. Common uses of Table storage include:

- Storing TBs of structured data capable of serving web scale applications
- Storing datasets that don't require complex joins, foreign keys, or stored procedures and can be denormalized for fast access
- Quickly querying data using a clustered index
- Accessing data using the OData protocol and LINQ queries with WCF Data Service .NET Libraries

You can use Table storage to store and query huge sets of structured, non-relational data, and your tables will scale as demand increases.

### 1.4.2 Table storage concepts



- **URL format:** Azure Table Storage accounts use this format: `http://<storage account>.table.core.windows.net/<table>`  
Azure Cosmos DB Table API accounts use this format: `http://<storage account>.table.cosmosdb.azure.com/<table>`  
You can address Azure tables directly using this address with the OData protocol.
- **Accounts:** All access to Azure Storage is done through a storage account.  
All access to Azure Cosmos DB is done through a Table API account.
- **Table:** A table is a collection of entities. Tables don't enforce a schema on entities, which means a single table can contain entities that have different sets of properties.
- **Entity:** An entity is a set of properties, similar to a database row. An entity in Azure Storage can be up to 1MB in size. An entity in Azure Cosmos DB can be up to 2MB in size.
- **Properties:** A property is a name-value pair. Each entity can include up to 252 properties to store data. Each entity also has three system properties that specify a partition key, a row key, and a timestamp. Entities with the same partition key can be queried more quickly, and inserted/updated in atomic operations. An entity's row key is its unique identifier within a partition.

## 1.5 Azure Disk storage

Azure managed disks are block-level storage volumes that are managed by Azure and used with Azure Virtual Machines. Managed disks are like a physical disk in an on-premises server but, virtualized. With managed disks, all you have to do is specify the disk size, the disk type, and provision the disk. Once you provision the disk, Azure handles the rest.

The available types of disks are ultra disks, premium solid-state drives (SSD), standard SSDs, and standard hard disk drives (HDD).

### **1.5.1. Benefits of managed disks**

Azure Managed Disks bring lots of advantages and benefits to the table when compared to Azure Storage Account based VM disks. The first thing to note is that these are only related to VM disks and not general blob storage. In this post, lets take a look at what all benefits you get when you create your virtual machine with a managed disk instead of a storage account based disks.

## **More Controlled Access Management**

Let's assume a scenario where all the disks for the virtual machines in your environment belonged to a particular storage account. Let us say that there are VMs belonging to Finance as well as HR departments. Now if you want to give access to a VHD file of a VM belonging to HR department, you will provide the access to the storage account. This was the lowest level where you could provide the access. This inadvertently opened the access to the Finance VM's VHD files as well.

Managed Disks are individual resources in Azure. If a VM has 1 OS disk and 2 data disks, all implemented as a managed disk, then you can even provide the access to one of the data disk and not provide access to any of the other disks.

## **No Storage account service limits**

Earlier with storage accounts there were Service Limits related to IOPS at the storage account level. When the infrastructure grew and there comes a time the number of disks grew to a point that this service limit will be hit and this can affect your architecture. With managed disks, you are no longer limited by the storage account limits.

## **Ability to take Snapshot**

Now with managed disks you have the capability to take snapshots on the fly. You can later restore from these snapshots as required. You can take these snapshots onto a different storage account.

## **Ability to Capture better images**

The images that you capture on the Vms, which are created using managed disks, will not just include the OS disk, but will also include all the data disk.

## **Ability to convert a Standard disk to Premium disk and vice versa**

Earlier if you wanted to convert a standard disk to a premium disk (or vice versa) you needed to create a new storage account and copy over the disk. Now with managed disks, this is as easy as shutting down the virtual machine and just changing a value in a drop down.

## **Other benefits**

Other benefits include:

- Better reliability for Availability Sets
- Highly durable and available with design for 99.999% availability
- Better Azure Backup service support with the ability to create a backup job with time-based backups, easy VM restoration, and backup retention policies.

## **1.5.2 Security**

### **Private Links**

Private Link support for managed disks can be used to import or export a managed disk internal to your network. Private Links allow you to generate a time bound Shared Access Signature (SAS) URI for unattached managed disks and snapshots that you can use to export the data to other regions for regional expansion, disaster recovery, and forensic analysis. You can also use the SAS URI to directly upload a VHD to an empty disk from on-premises. Now you can leverage Private links to restrict the export and import of managed disks so that it can only occur within your Azure virtual network. Private Links allows you to ensure your data only travels within the secure Microsoft backbone network.

### **Encryption**

Managed disks offer two different kinds of encryption. The first is Server-Side Encryption (SSE), which is performed by the storage service. The second one is Azure Disk Encryption (ADE), which you can enable on the OS and data disks for your VMs.

#### **Server-side encryption**

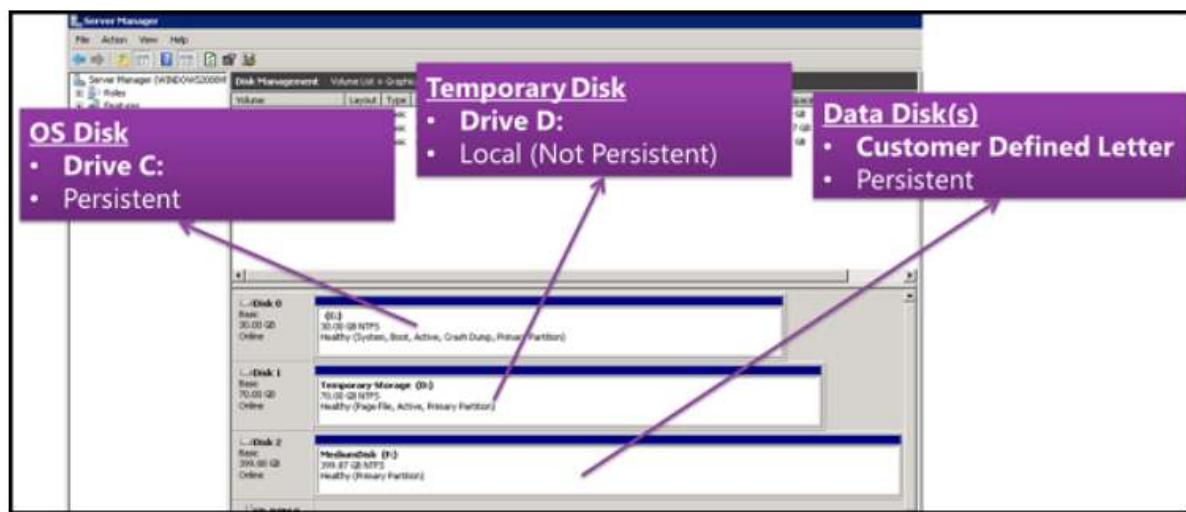
Server-side encryption provides encryption-at-rest and safeguards your data to meet your organizational security and compliance commitments. Server-side encryption is enabled by default for all managed disks, snapshots, and images, in all the regions where managed disks are available.

You can either allow Azure to manage your keys for you, these are platform-managed keys, or you can manage the keys yourself, these are customer-managed keys.

### Azure Disk Encryption

Azure Disk Encryption allows you to encrypt the OS and Data disks used by an IaaS Virtual Machine. This encryption includes managed disks. For Windows, the drives are encrypted using industry-standard BitLocker encryption technology. For Linux, the disks are encrypted using the DM-Crypt technology. The encryption process is integrated with Azure Key Vault to allow you to control and manage the disk encryption keys.

### 1.5.3 Disk Roles



#### Data disk

A data disk is a managed disk that's attached to a virtual machine to store application data, or other data you need to keep. Data disks are registered as SCSI drives and are labelled with a letter that you choose. Each data disk has a maximum capacity of 32,767 gibibytes (GiB). The size of the virtual machine determines how many data disks you can attach to it and the type of storage you can use to host the disks.

#### OS disk

Every virtual machine has one attached operating system disk. That OS disk has a pre-installed OS, which was selected when the VM was created. This disk contains the boot volume. This disk has a maximum capacity of 4,095 GiB.

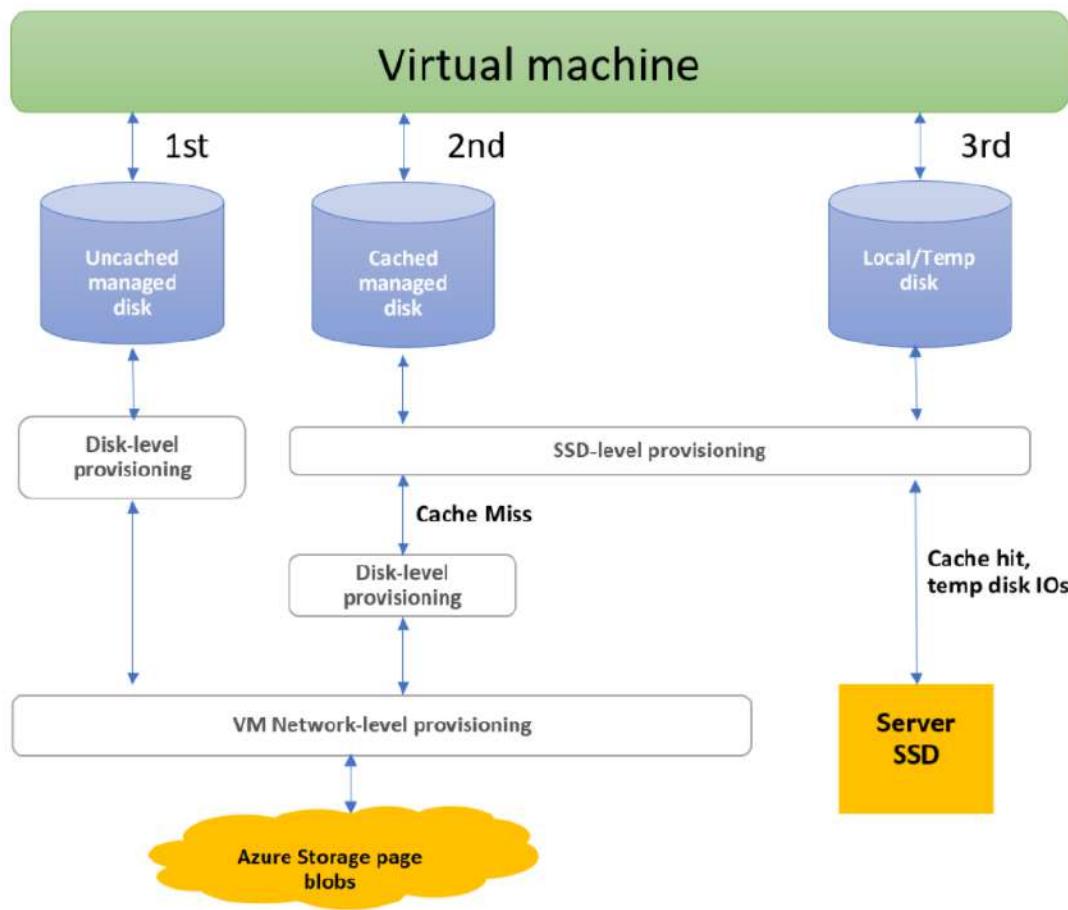
#### Temporary disk

Most VMs contain a temporary disk, which is not a managed disk. The temporary disk provides short-term storage for applications and processes, and is intended to only store

data such as page or swap files. Data on the temporary disk may be lost during a maintenance event or when you redeploy a VM. During a successful standard reboot of the VM, data on the temporary disk will persist.

On Azure Linux VMs, the temporary disk is typically /dev/sdb and on Windows VMs the temporary disk is D: by default. The temporary disk is not encrypted by server side encryption unless you enable encryption at host.

#### 1.5.4 Disk allocation and performance

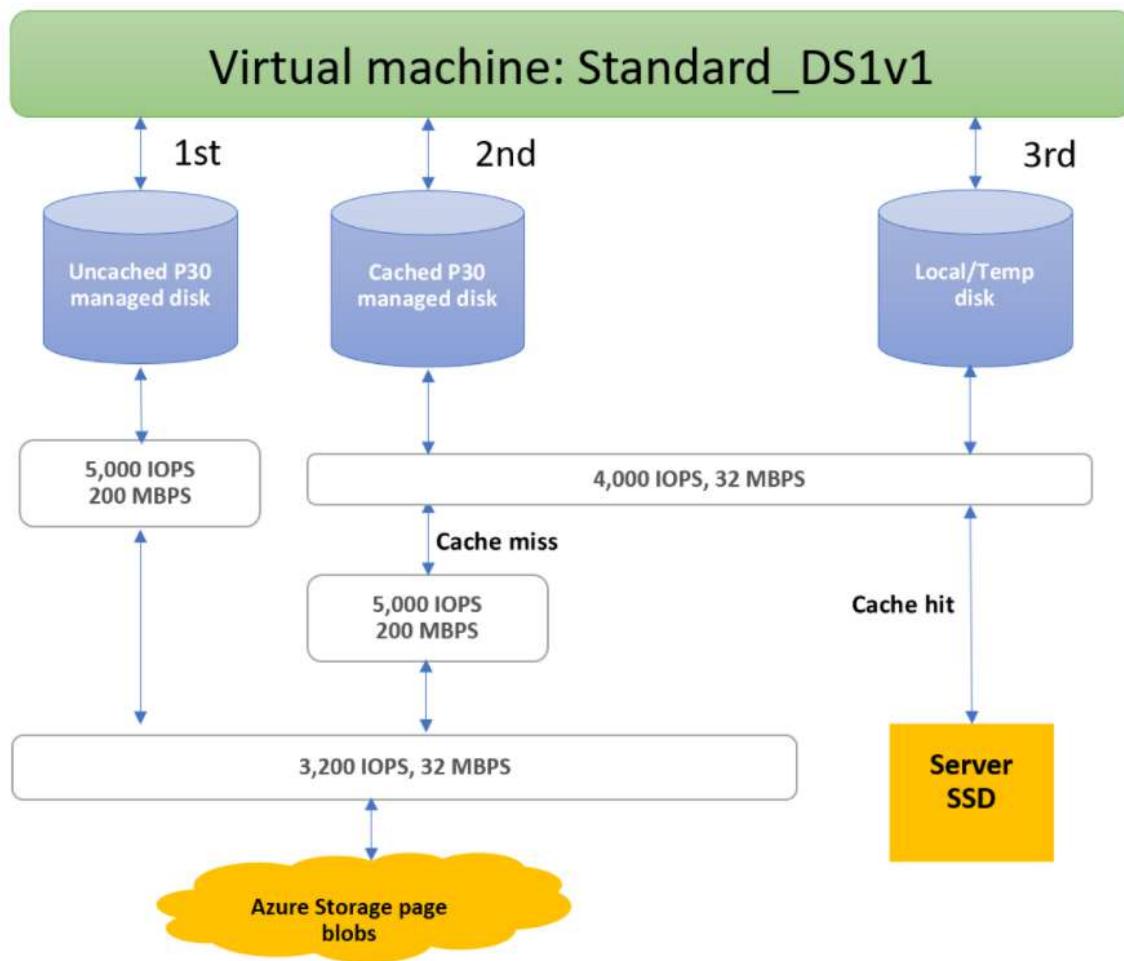


1. The first IO path is the uncached managed disk path. This path is taken if you are using a managed disk and set the host caching to none. An IO using this path will execute based on disk-level provisioning and then VM network-level provisioning for IOPs and throughput.
2. The second IO Path is the cached managed disk path. Cached managed disk IO uses an SSD close to the VM, which have their own IOPs and throughput provisioned, and is labeled SSD-level provisioning in the diagram. When a cached managed disk initiates a read, the request first checks to see if the data is in the server SSD. If the data isn't present, this creates a cached miss and the IO then executes based on SSD-level provisioning, disk-level provisioning and then VM network-level provisioning

for IOPs and throughput. When the server SSD initiates reads on cached IO that are present on the server SSD, it creates a cache hit and the IO will then execute based on the SSD-level provisioning. Writes initiated by a cached managed disk always follow the path of a cached-miss, and need to go through SSD-level, disk-level, and VM network-level provisioning.

3. Finally, the third path is for the local/temp disk. This is available only on VMs that support local/temp disks. An IO using this path will execute based on SSD-Level Provisioning for IOPs and throughput.

As an example of these limitations, a Standard\_DS1v1 VM is prevented from achieving the 5,000 IOPS potential of a P30 disk, whether it is cached or not, because of limits at the SSD and network levels:



Azure uses prioritized network channel for disk traffic, which gets the precedence over other low priority of network traffic. This helps disks maintain their expected performance in case of network contentions. Similarly, Azure Storage handles resource contentions and other issues in the background with automatic load balancing. Azure Storage allocates required resources when you create a disk, and applies proactive and reactive balancing of resources to handle the traffic level. This further ensures disks can sustain their

expected IOPS and throughput targets. You can use the VM-level and Disk-level metrics to track the performance and setup alerts as needed.

### 1.5.5 Managed disk snapshots

A managed disk snapshot is a read-only crash-consistent full copy of a managed disk that is stored as a standard managed disk by default. With snapshots, you can back up your managed disks at any point in time. These snapshots exist independent of the source disk and can be used to create new managed disks.

Snapshots are billed based on the used size. For example, if you create a snapshot of a managed disk with provisioned capacity of 64 GiB and actual used data size of 10 GiB, that snapshot is billed only for the used data size of 10 GiB.

You can see the used size of your snapshots by looking at the Azure usage report. For example, if the used data size of a snapshot is 10 GiB, the daily usage report will show  $10 \text{ GiB} / (31 \text{ days}) = 0.3226$  as the consumed quantity.

## Images

Managed disks also support creating a managed custom image. You can create an image from your custom VHD in a storage account or directly from a generalized (sysprepped) VM. This process captures a single image. This image contains all managed disks associated with a VM, including both the OS and data disks. This managed custom image enables creating hundreds of VMs using your custom image without the need to copy or manage any storage accounts.

## Images versus snapshots

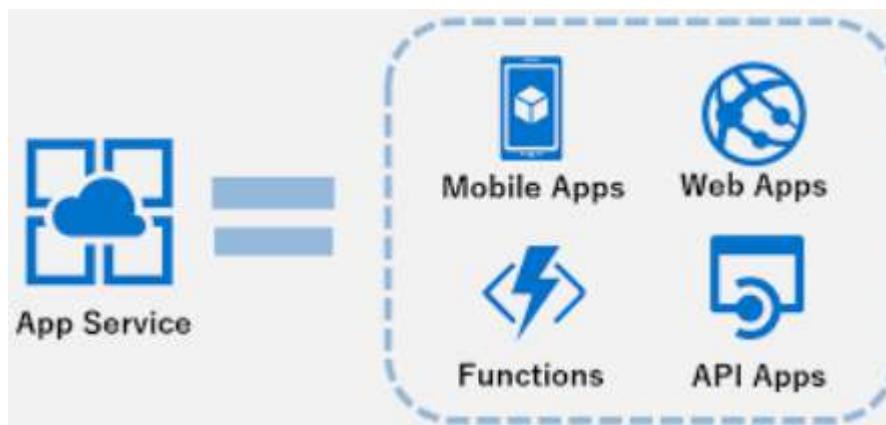
1. It's important to understand the difference between images and snapshots. With managed disks, you can take an image of a generalized VM that has been deallocated. This image includes all of the disks attached to the VM. You can use this image to create a VM, and it includes all of the disks.
2. A snapshot is a copy of a disk at the point in time the snapshot is taken. It applies only to one disk. If you have a VM that has one disk (the OS disk), you can take a snapshot or an image of it and create a VM from either the snapshot or the image.
3. A snapshot doesn't have awareness of any disk except the one it contains. This makes it problematic to use in scenarios that require the coordination of multiple disks, such as striping. Snapshots would need to be able to coordinate with each other and this is currently not supported.

## **AZURE COMPUTE DOCUMENTATION**

### **AZURE APP SERVICE**

Azure App Service enables you to build and host web apps, mobile back ends, and RESTful APIs in the programming language of your choice without managing infrastructure. It offers auto-scaling and high availability, supports both Windows and Linux, and enables automated deployments from GitHub, Azure DevOps, or any Git repo. Learn how to use Azure App Service with our quickstarts, tutorials, and samples.

Azure App Service is an HTTP-based service for hosting web applications, REST APIs, and mobile back ends. You can develop in your favorite language, be it .NET, .NET Core, Java, Ruby, Node.js, PHP, or Python. Applications run and scale with ease on both Windows and Linux-based environments.

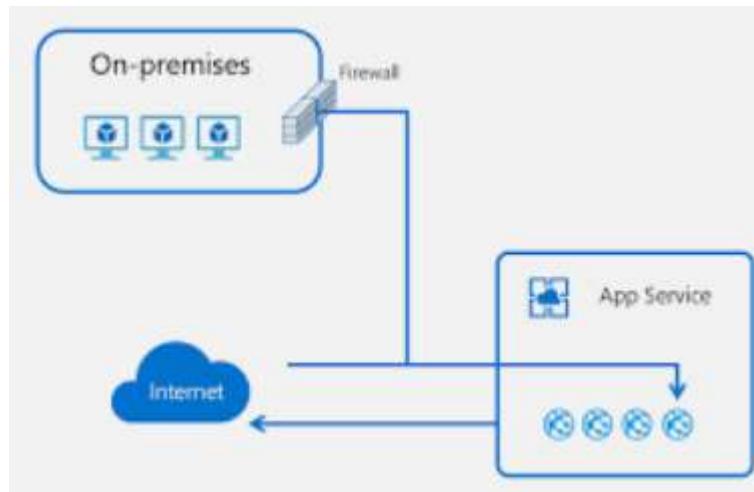


App Service not only adds the power of Microsoft Azure to your application, such as security, load balancing, autoscaling, and automated management. You can also take advantage of its DevOps capabilities, such as continuous deployment from Azure DevOps, GitHub, Docker Hub, and other sources, package management, staging environments, custom domain, and TLS/SSL certificates.

With App Service, you pay for the Azure compute resources you use. The compute resources you use are determined by the App Service plan that you run your apps on. For more information, see [Azure App Service plans overview](#).

## FEATURES OF APP SERVICE

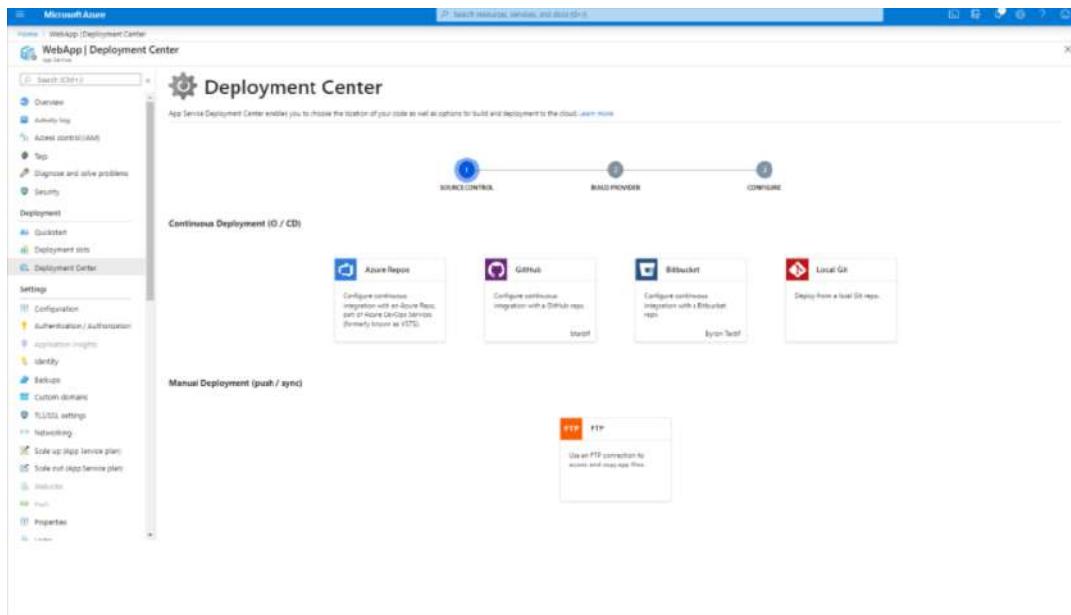
- Multiple languages and frameworks - App Service has first-class support for ASP.NET, ASP.NET Core, Java, Ruby, Node.js, PHP, or Python. You can also run PowerShell and other scripts or executables as background services.
- Managed production environment - App Service automatically patches and maintains the OS and language frameworks for you. Spend time writing great apps and let Azure worry about the platform.
- Containerization and Docker - Dockerize your app and host a custom Windows or Linux container in App Service. Run multi-container apps with Docker Compose. Migrate your Docker skills directly to App Service.
- DevOps optimization - Set up continuous integration and deployment with Azure DevOps, GitHub, BitBucket, Docker Hub, or Azure Container Registry. Promote updates through test and staging environments. Manage your apps in App Service by using Azure PowerShell or the cross-platform command-line interface (CLI).
- Global scale with high availability - Scale up or out manually or automatically. Host your apps anywhere in Microsoft's global datacenter infrastructure, and the App Service SLA promises high availability.
- Connections to SaaS platforms and on-premises data - Choose from more than 50 connectors for enterprise systems (such as SAP), SaaS services (such as Salesforce), and internet services (such as Facebook). Access on-premises data using Hybrid Connections and Azure Virtual Networks.
- Security and compliance - App Service is ISO, SOC, and PCI compliant. Authenticate users with Azure Active Directory, Google, Facebook, Twitter, or Microsoft account. Create IP address restrictions and manage service identities.
- Application templates - Choose from an extensive list of application templates in the Azure Marketplace, such as WordPress, Joomla, and Drupal.
- Visual Studio and Visual Studio Code integration - Dedicated tools in Visual Studio and Visual Studio Code streamline the work of creating, deploying, and debugging.
- API and mobile features - App Service provides turn-key CORS support for RESTful API scenarios, and simplifies mobile app scenarios by enabling authentication, offline data sync, push notifications, and more.
- Serverless code - Run a code snippet or script on-demand without having to explicitly provision or manage infrastructure, and pay only for the compute time your code actually uses (see Azure Functions).



## QUICKLY BUILD WEB APPS AND APIS IN THE CLOUD

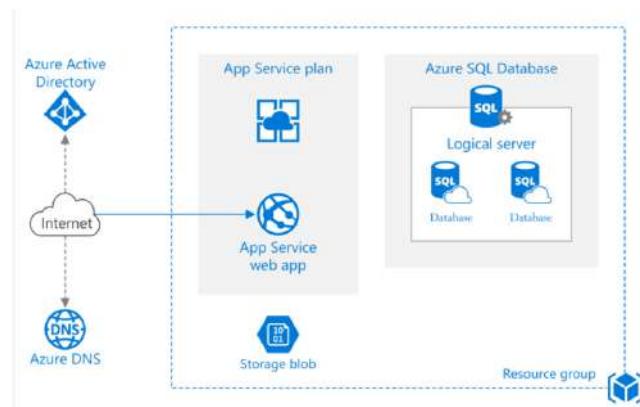
- Bring your code or container using the framework language of your choice.
- Run on Kubernetes, anywhere across Azure, on-premises, and any CNCF-conformant Kubernetes cluster through Azure Arc.
- Increase developer productivity with tight integration of Visual Studio Code and Visual Studio.
- Streamline CI/CD with Git, GitHub, GitHub Actions, Atlassian Bitbucket, Azure DevOps, Docker Hub, and Azure Container Registry.
- Reduce downtime and minimize risk for app updates by using deployment slots.

Besides App Service, Azure offers other services that can be used for hosting websites and web applications. For most scenarios, App Service is the best choice. For microservice architecture, consider Azure Spring-Cloud Service or Service Fabric. If you need more control over the VMs on which your code runs, consider Azure Virtual Machines. For more information about how to choose between these Azure services, see [Azure App Service, Virtual Machines, Service Fabric, and Cloud Services comparison](#).



## SCALE WEB APPS ON AN ENTERPRISE-GRADE SERVICE

- Get high availability with a service-level agreement (SLA)-backed uptime of 99.95 percent.
- Simplify operations with automatic platform maintenance and security patching.
- Help protect your applications with Azure Web Application Firewall, and connect through virtual network integration.
- Deploy isolated web app instances with a single-tenancy model. Use App Service Environment v3 to enforce network access external to your applications.
- Use Azure Active Directory and other popular identity providers to authenticate and authorize app access.
- Scale globally across all Azure regions.



## SCALE GLOBALLY WITH INDUSTRY-LEADING SECURITY

- React to traffic loads with Azure Autoscale, and perform traffic routing and load-balancing through Azure Front Door.
- Reduce latency by placing your content assets closer to your customers, with Azure Content Delivery Network.
- Help protect your applications with Azure Web Application Firewall, Azure Firewall, and Application Gateway.
- Improve your security posture and threat protection with Azure Security Center and extensive compliance certifications.

The screenshot shows the Microsoft Azure Security Center - Compute & apps dashboard. On the left, there's a navigation sidebar with various service icons like Home, Dashboard, All services, Favorites, All resources, Resource groups, App Services, Function Apps, SQL databases, Azure Cosmos DB, Virtual machines, Load balancers, Storage accounts, Virtual networks, Azure Active Directory, Monitor, Advisor, Security Center, and Cost Management + Billing. The main area is titled "Security Center - Compute & apps" and shows "Showing 2 subscriptions". It has sections for "Events" and "Search". Below these are "POLICY & COMPLIANCE" and "RECOMMENDATION". The "RECOMMENDATION" section lists several items with progress bars indicating completion. For example, "Remediate vulnerabilities in container security configurations" is at 74%, "Enable Adaptive Application Controls" is at 25%, and "Install system updates on your machines" is at 15%. Other items include setting ClusterProtectionLevel, applying disk encryption, installing endpoint protection, using Azure Active Directory for client authentication, installing endpoint protection on machines, remediating vulnerabilities via Vulnerability Assessment, installing a vulnerability assessment solution, and ensuring web applications are accessible over HTTPS.

RECOMMENDATION	SECURE SCO...	RESOURCE
Remediate vulnerabilities in container security configurations	+35	1 of 1 Container h...
Enable Adaptive Application Controls	+25	4 of 104 virtual m...
Install system updates on your machines	+15	23 of 111 VMs & c...
Set the 'ClusterProtectionLevel' property to 'EncryptAndSign'	+15	1 of 1 service fabr...
Apply disk encryption on your virtual machines	+11	84 of 104 virtual ...
Install endpoint protection solution on virtual machines	+10	27 of 104 virtual ...
Use Azure Active Directory for client authentication (Preview)	+10	1 of 1 service fabr...
Install endpoint protection solution on your machines	+7	1 of 7 computers
Remediate vulnerabilities - by a Vulnerability Assessment col...	+5	1 of 104 virtual m...
Install a vulnerability assessment solution on your virtual mac...	+29	68 of 104 virtual ...
Web Application should only be accessible over HTTPS	+29	10 of 10 web appli...

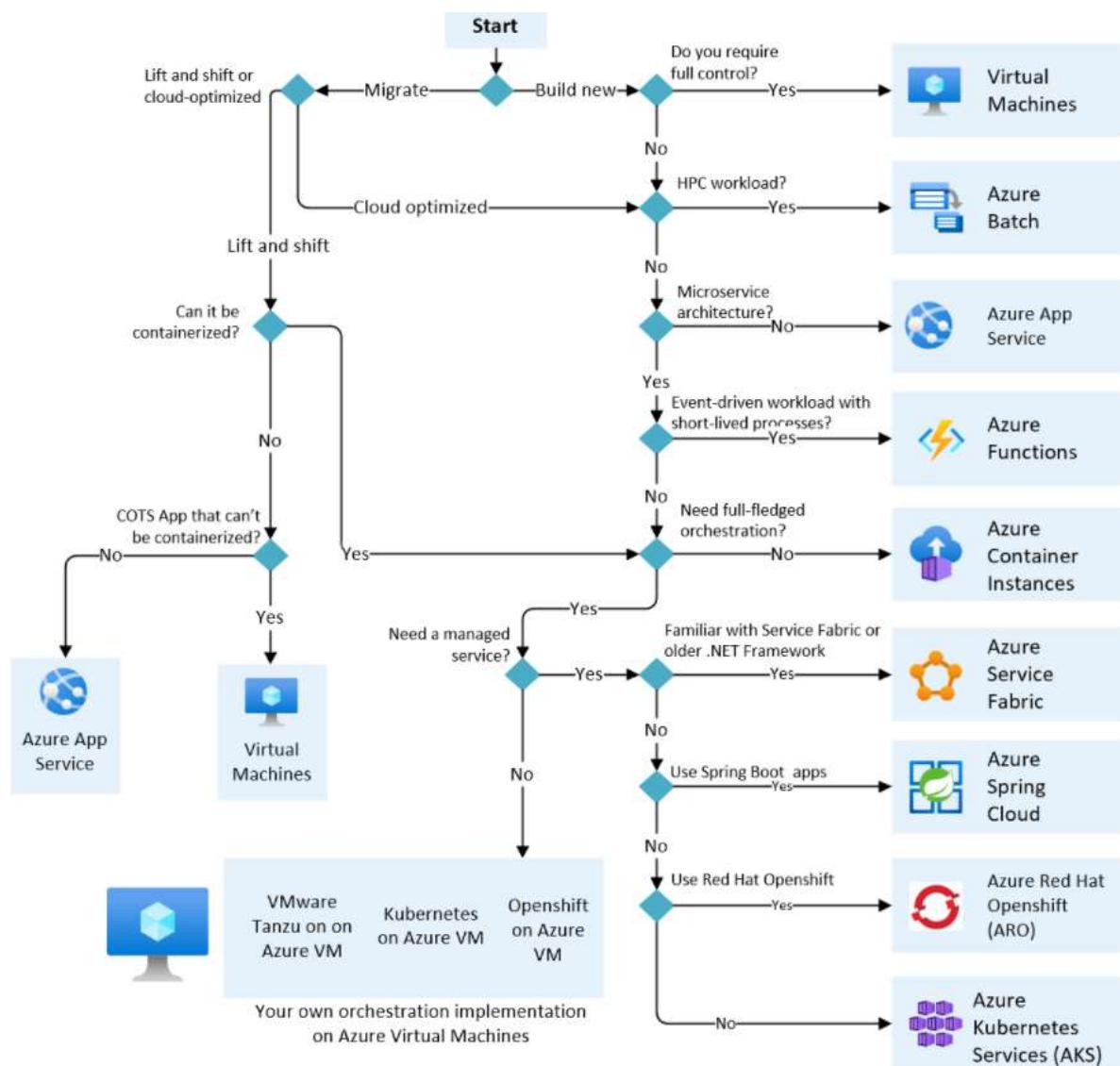
## CHOOSE AN AZURE COMPUTE SERVICE FOR YOUR APPLICATION

Azure offers a number of ways to host your application code. The term compute refers to the hosting model for the computing resources that your application runs on. The following flowchart will help you to choose a compute service for your application.

If your application consists of multiple workloads, evaluate each workload separately. A complete solution may incorporate two or more compute services.

## CHOOSE A CANDIDATE SERVICE

Use the following flowchart to select a candidate compute service.



## DEFINITIONS:

- "Lift and shift" is a strategy for migrating a workload to the cloud without redesigning the application or making code changes. Also called rehosting. For more information, see Azure migration and modernization center.
- Cloud optimized is a strategy for migrating to the cloud by refactoring an application to take advantage of cloud-native features and capabilities.

The output from this flowchart is a starting point for consideration. Next, perform a more detailed evaluation of the service to see if it meets your needs.

## FEATURES

- App Service. A managed service for hosting web apps, mobile app back ends, RESTful APIs, or automated business processes.
- Azure Spring Cloud. A managed service designed and optimized for hosting Spring Boot apps.
- Azure Kubernetes Service (AKS). A managed Kubernetes service for running containerized applications.
- Batch. A managed service for running large-scale parallel and high-performance computing (HPC) applications
- Container Instances. The fastest and simplest way to run a container in Azure, without having to provision any virtual machines and without having to adopt a higher-level service.
- Functions. A managed FaaS service.
- Service Fabric. A distributed systems platform that can run in many environments, including Azure or on premises.
- Virtual machines. Deploy and manage VMs inside an Azure virtual network.

## HOSTING MODELS

Cloud services, including Azure services, generally fall into three categories: IaaS, PaaS, or FaaS. (There is also SaaS, software-as-a-service, which is out of scope for this article.) It's useful to understand the differences.

**Infrastructure-as-a-Service (IaaS)** lets you provision individual VMs along with the associated networking and storage components. Then you deploy whatever software and applications you want onto those VMs. This model is the closest to a traditional on-premises environment, except that Microsoft manages the infrastructure. You still manage the individual VMs.

**Platform-as-a-Service (PaaS)** provides a managed hosting environment, where you can deploy your application without needing to manage VMs or networking resources. Azure App Service is a PaaS service.

**Functions-as-a-Service (FaaS)** goes even further in removing the need to worry about the hosting environment. In a FaaS model, you simply deploy your code and the service automatically runs it. Azure Functions is a FaaS service.

There is a spectrum from IaaS to pure PaaS. For example, Azure VMs can autoscale by using virtual machine scale sets. This automatic scaling capability isn't strictly PaaS, but it's the type of management feature found in PaaS services.

In general, there is a tradeoff between control and ease of management. IaaS gives the most control, flexibility, and portability, but you have to provision, configure and manage the VMs and network components you create. FaaS services automatically manage nearly all aspects of running an application. PaaS services fall somewhere in between.

## DEVOPS

Criteria	Virtual Machines	App Service	Azure Spring Cloud	Service Fabric	Azure Functions	Azure Kubernetes Service	Container Instances	Azure Batch
Local debugging	Agnostic	IIS Express, others	Visual Studio Code, IntelliJ, Eclipse	Local node cluster	Visual Studio or Azure Functions CLI	Minikube, others	Local container runtime	Not supported
Programming model	Agnostic	Web and API applications, WebJobs for background tasks	Spring Boot, Steeltoe	Guest executable, Service model, Actor model, Containers	Functions with triggers	Agnostic	Agnostic	Command line application
Application update	No built-in support	Deployment slots	Rolling upgrade, Blue-green deployment	Rolling upgrade (per service)	Deployment slots	Rolling update	Not applicable	

## SCALABILITY

Criteria	Virtual Machines	App Service	Azure Spring Cloud	Service Fabric	Azure Functions	Azure Kubernetes Service	Container Instances	Azure Batch
Autoscaling	Virtual machine scale sets	Built-in service	Built-in service	Virtual machine scale sets	Built-in service	Pod auto-scaling, cluster auto-scaling	Not supported	N/A
Load balancer	Azure Load Balancer	Integrated	Integrated	Azure Load Balancer	Integrated	Azure Load Balancer or Application Gateway	No built-in support	Azure Load Balancer
Scale limit	Platform image: 1000 nodes per scale set, Custom image: 600 nodes per scale set	30 instances, 100 with App Service Environment	500 app instances in Standard	100 nodes per scale set	200 instances per Function app	100 nodes per cluster (default limit)	20 container groups per subscription (default limit).	20 core limits (default limit).

## AVAILABILITY

Criteria	Virtual Machines	App Service	Azure Spring Cloud	Service Fabric	Azure Functions	Azure Kubernetes Service	Container Instances	Azure Batch
SLA	SLA for Virtual Machines	SLA for App Service	SLA for Azure Spring Cloud	SLA for Service Fabric	SLA for Functions	SLA for AKS	SLA for Container Instances	SLA for Azure Batch
Multi region failover	Traffic manager	Traffic manager	Traffic manager, Multi-Region Cluster	Azure Front Door	Traffic manager	Not supported	Not Supported	

## OTHER CRITERIA

Criteria	Virtual Machines	App Service	App Spring Cloud	Service Fabric	Azure Functions	Azure Kubernetes Service	Container Instances	Azure Batch
SSL	Configured in VM	Supported	Supported	Supported	Supported	Ingress controller	Use sidecar container	Supported
Cost	Windows, Linux	App Service pricing	Azure Spring Cloud pricing	Service Fabric pricing	Azure Functions pricing	AKS pricing	Container Instances pricing	Azure Batch pricing
Suitable architecture styles	N-Tier, Big compute (HPC)	Web-Queue-Worker, N-Tier	Spring Boot, Microservices	Microservices, Event-driven architecture	Microservices, Event-driven architecture	Microservices, Event-driven architecture	Microservices, task automation, batch jobs	Big compute (HPC)

## APP SERVICE (LINUX)

App Service can also host web apps natively on Linux for supported application stacks. It can also run custom Linux containers (also known as Web App for Containers).

## BUILT-IN LANGUAGES AND FRAMEWORKS

App Service on Linux supports a number of language specific built-in images. Just deploy your code. Supported languages include: Node.js, Java (JRE 8 & JRE 11), PHP, Python, .NET Core, and Ruby. Run `az webapp list-runtimes --linux` to view the latest languages and supported versions. If the runtime your application requires is not supported in the built-in images, you can deploy it with a custom container.

Outdated runtimes are periodically removed from the Web Apps Create and Configuration blades in the Portal. These runtimes are hidden from the Portal when they are deprecated by the maintaining organization or found to have significant vulnerabilities. These

options are hidden to guide customers to the latest runtimes where they will be the most successful.

When an outdated runtime is hidden from the Portal, any of your existing sites using that version will continue to run. If a runtime is fully removed from the App Service platform, your Azure subscription owner(s) will receive an email notice before the removal.

If you need to create another web app with an outdated runtime version that is no longer shown on the Portal see the language configuration guides for instructions on how to get the runtime version of your site. You can use the Azure CLI to create another site with the same runtime. Alternatively, you can use the Export Template button on the web app blade in the Portal to export an ARM template of the site. You can reuse this template to deploy a new site with the same runtime and configuration.

## LIMITATIONS

- App Service on Linux is not supported on Shared pricing tier.
- The Azure portal shows only features that currently work for Linux apps. As features are enabled, they're activated on the portal.
- When deployed to built-in images, your code and content are allocated a storage volume for web content, backed by Azure Storage. The disk latency of this volume is higher and more variable than the latency of the container filesystem. Apps that require heavy read-only access to content files may benefit from the custom container option, which places files in the container filesystem instead of on the content volume.

## AZURE BATCH

Azure Batch runs large-scale applications efficiently in the cloud. Schedule compute-intensive tasks and dynamically adjust resources for your solution without managing infrastructure.

Developers can use Batch as a platform service to build SaaS applications or client apps where large-scale execution is required. For example, you can build a service with Batch to run a Monte Carlo risk simulation for a financial services company, or a service to process many images.

There is no additional charge for using Batch. You only pay for the underlying resources consumed, such as the virtual machines, storage, and networking.

## RUN PARALLEL WORKLOAD

Batch works well with intrinsically parallel (also known as "embarrassingly parallel") workloads. These workloads have applications which can run independently, with each instance completing part of the work. When the applications are executing, they might access some common data, but they don't communicate with other instances of the application. Intrinsically parallel workloads can therefore run at a large scale, determined by the amount of compute resources available to run applications simultaneously.

Some examples of intrinsically parallel workloads you can bring to Batch:

- Financial risk modeling using Monte Carlo simulations
- VFX and 3D image rendering
- Image analysis and processing
- Media transcoding
- Genetic sequence analysis
- Optical character recognition (OCR)
- Data ingestion, processing, and ETL operations
- Software test execution

You can also use Batch to run tightly coupled workloads, where the applications you run need to communicate with each other, rather than running independently. Tightly coupled applications normally use the Message Passing Interface (MPI) API. You can run your tightly coupled workloads with Batch using Microsoft MPI or

Intel MPI. Improve application performance with specialized HPC and GPU-optimized VM sizes.

- Some examples of tightly coupled workloads:
- Finite element analysis
- Fluid dynamics
- Multi-node AI training

Many tightly coupled jobs can be run in parallel using Batch. For example, you can perform multiple simulations of a liquid flowing through a pipe with varying pipe widths.

## **ADDITIONAL BATCH CAPABILITIES**

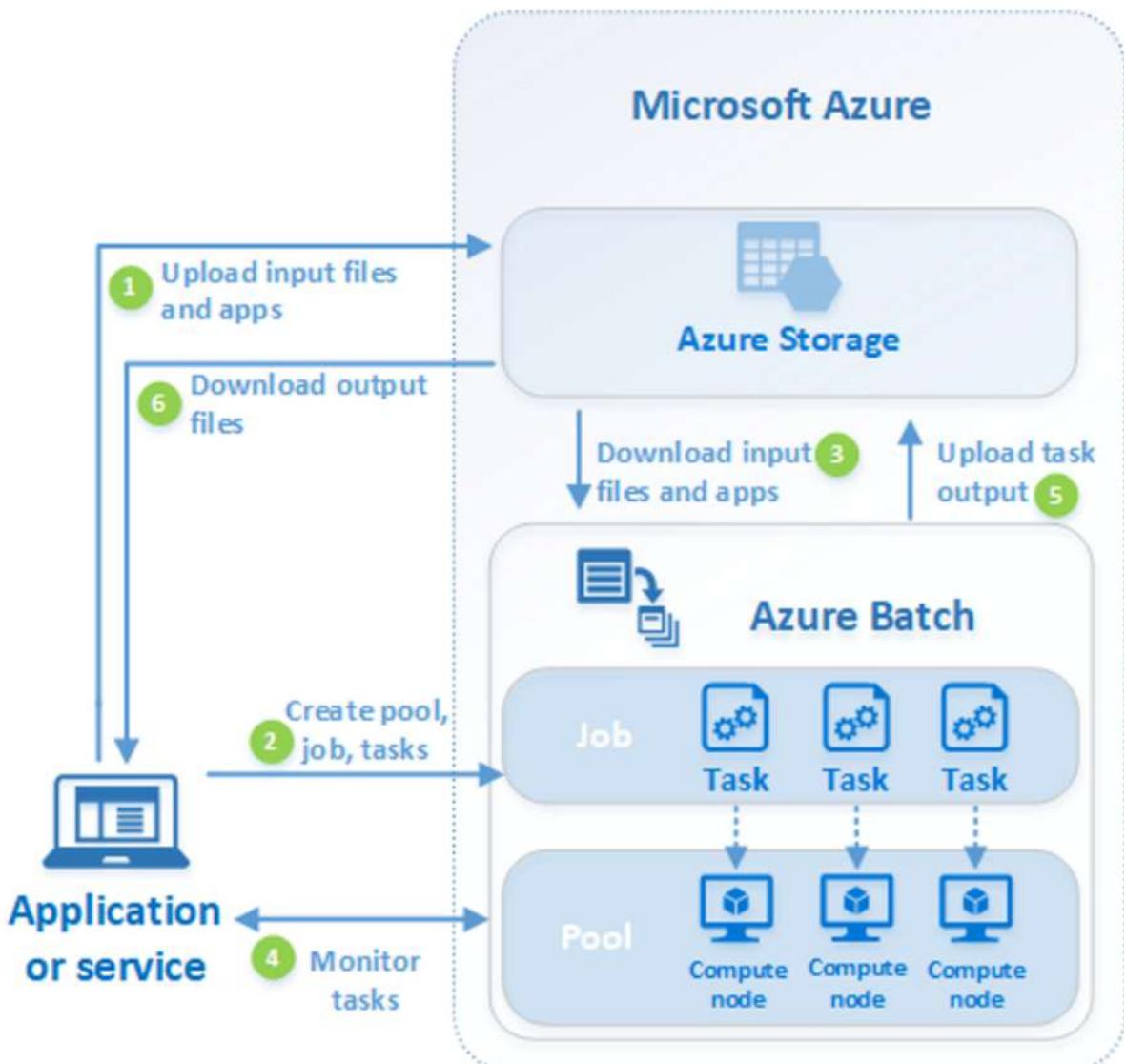
Batch supports large-scale rendering workloads with rendering tools including Autodesk Maya, 3ds Max, Arnold, and V-Ray.

You can also run Batch jobs as part of a larger Azure workflow to transform data, managed by tools such as Azure Data Factory.

## **HOW IT WORKS**

A common scenario for Batch involves scaling out intrinsically parallel work, such as the rendering of images for 3D scenes, on a pool of compute nodes. This pool can be your "render farm" that provides tens, hundreds, or even thousands of cores to your rendering job.

The following diagram shows steps in a common Batch workflow, with a client application or hosted service using Batch to run a parallel workload.



Step	Description
1. Upload <b>input files</b> and the <b>applications</b> to process those files to your Azure Storage account.	The input files can be any data that your application processes, such as financial modeling data, or video files to be transcoded. The application files can include scripts or applications that process the data, such as a media transcoder.
2. Create a Batch <b>pool</b> of compute nodes in your Batch account, a <b>job</b> to run the workload on the pool, and <b>tasks</b> in the job.	Compute <b>nodes</b> are the VMs that execute your <b>tasks</b> . Specify properties for your pool, such as the number and size of the nodes, a Windows or Linux VM image, and an application to install when the nodes join the pool. Manage the cost and size of the pool by using <a href="#">Azure Spot VMs</a> or by automatically scaling the number of nodes as the workload changes.
	When you add tasks to a job, the Batch service automatically schedules the tasks for execution on the compute nodes in the pool. Each task uses the application that you uploaded to process the input files.
3. Download <b>input files</b> and the <b>applications</b> to Batch	Before each task executes, it can download the input data that it will process to the assigned node. If the application isn't already installed on the pool nodes, it can be downloaded here instead. When the downloads from Azure Storage complete, the task executes on the assigned node.
4. Monitor <b>task execution</b>	As the tasks run, query Batch to monitor the progress of the job and its tasks. Your client application or service communicates with the Batch service over HTTPS. Because you may be monitoring thousands of tasks running on thousands of compute nodes, be sure to <a href="#">query the Batch service efficiently</a> .
5. Upload <b>task output</b>	As the tasks complete, they can upload their result data to Azure Storage. You can also retrieve files directly from the file system on a compute node.
6. Download <b>output files</b>	When your monitoring detects that the tasks in your job have completed, your client application or service can download the output data for further processing.

## AZURE FUNCTIONS

Azure Functions is a cloud service available on-demand that provides all the continually updated infrastructure and resources needed to run your applications. Functions provides serverless compute for Azure.

Azure Functions is a serverless solution that allows you to write less code, maintain less infrastructure, and save on costs. Instead of worrying about deploying and maintaining servers, the cloud infrastructure provides all the up-to-date resources needed to keep your applications running.

Whether building a web API, responding to database changes, processing IoT data streams, or even managing message queues - every application needs a way to run some code as these events occur.

To meet this need, Azure Functions provides "compute on-demand" in two significant ways.

First, Azure Functions allows you to implement your system's logic into readily available blocks of code. These code blocks are called "functions". Different functions can run anytime you need to respond to critical events.

Second, as requests increase, Azure Functions meets the demand with as many resources and function instances as necessary - but only while needed. As requests fall, any extra resources and application instances drop off automatically.

## CREATE A FUNCTION TO INTEGRATE WITH AZURE LOGIC APPS

### Create Text Analytics resource

The Cognitive Services APIs are available in Azure as individual resources. Use the Text Analytics API.

1. Sign in to the Azure portal.
2. Select Create a resource in the upper left-hand corner of the Azure portal.
3. Under Categories, select AI + Machine Learning
4. Under Text Analytics, select Create.
5. Enter the following values in the Create Text Analytics screen.

Setting	Value	Remarks
Subscription	Your Azure subscription name	
Resource group	Create a new resource group named <b>tweet-sentiment-tutorial</b>	Later, you delete this resource group to remove all the resources created during this tutorial.
Region	Select the region closest to you	
Name	<b>TweetSentimentApp</b>	

<b>Setting</b>	<b>Value</b>	<b>Remarks</b>
Pricing	Select <b>Free F0</b>	

6. Select Review + create.
7. Select Create.
8. Once the deployment is complete, select Go to Resource.

## Get Text Analytics settings

With the Text Analytics resource created, you'll copy a few settings and set them aside for later use.

1. Select Keys and Endpoint.
2. Copy Key 1 by clicking on the icon at the end of the input box.
3. Paste the value into a text editor.
4. Copy the Endpoint by clicking on the icon at the end of the input box.
5. Paste the value into a text editor.

## Create the function app

1. From the top search box, search for and select Function app.
2. Select Create.
3. Enter the following values.

<b>Setting</b>	<b>Suggested Value</b>	<b>Remarks</b>
Subscription	Your Azure subscription name	
Resource group	<b>tweet-sentiment-tutorial</b>	Use the same resource group name throughout this tutorial.

Setting	Suggested Value	Remarks
Function App name	<b>TweetSentimentAPI</b> + a unique suffix	Function application names are globally unique. Valid characters are <code>a-z</code> (case insensitive), <code>0-9</code> , and <code>-</code> .
Publish	<b>Code</b>	
Runtime stack	<b>.NET</b>	The function code provided for you is in C#.
Version	Select the latest version number	
Region	Select the region closest to you	

4. Select Review + create.
5. Select Create.
6. Once the deployment is complete, select Go to Resource.

## Create an HTTP-triggered function

1. From the left menu of the Functions window, select Functions.
2. Select Add from the top menu and enter the following values.

Setting	Value	Remarks
Development environment	<b>Develop in portal</b>	
Template	<b>HTTP Trigger</b>	
New Function	<b>TweetSentimentFunction</b>	This is the name of your function.
Authorization level	<b>Function</b>	

3. Select the Add button.

4. Select the Code + Test button.
5. Paste the following code in the code editor window.

```
#r "Newtonsoft.Json"

using System;
using System.Net;
using Microsoft.AspNetCore.Mvc;
using Microsoft.Extensions.Logging;
using Microsoft.Extensions.Primitives;
using Newtonsoft.Json;

public static async Task<IActionResult> Run(HttpContext req, ILogger log)
{
    string requestBody = String.Empty;
    using (StreamReader streamReader = new StreamReader(req.Body))
    {
        requestBody = await streamReader.ReadToEndAsync();
    }

    dynamic score = JsonConvert.DeserializeObject(requestBody);
    string value = "Positive";

    if(score < .3)
    {
        value = "Negative";
    }
}
```

```

    }

else if (score < .6)

{
    value = "Neutral";

}

return requestBody != null
    ? (ActionResult)new OkObjectResult(value)
    : new BadRequestObjectResult("Pass a sentiment score in the request body.");
}

```

6. Select the Save button on the toolbar to save your changes.

## Create a logic app

1. From the top search box, search for and select Logic Apps.
2. Select Add.
3. Select Consumption and enter the following values.

<b>Setting</b>	<b>Suggested Value</b>
Subscription	Your Azure subscription name
Resource group	<b>tweet-sentiment-tutorial</b>
Logic app name	<b>TweetSentimentApp</b>
Region	Select the region closest to you, preferably the same region you selected in previous steps.

4. Select Review + create.
5. Select Create.
6. Once the deployment is complete, select Go to Resource.

7. Select the Blank Logic App button.

Blank Logic App



8. Select the Save button on the toolbar to save your progress.

## Connect to Twitter

Create a connection to Twitter so your app can poll for new tweets.

1. Search for Twitter in the top search box.
2. Select the Twitter icon.
3. Select the When a new tweet is posted trigger.
4. Enter the following values to set up the connection.

Setting	Value
Connection name	<b>MyTwitterConnection</b>
Authentication Type	<b>Use default shared application</b>

5. Select Sign in.
6. Follow the prompts in the pop-up window to complete signing in to Twitter.
7. Next, enter the following values in the When a new tweet is posted box.

Setting	Value
Search text	<b>#my-twitter-tutorial</b>
How often do you want to check for items?	<b>1</b> in the textbox, and <b>Hour</b> in the dropdown. You may enter different values but be sure to review the current limitations of the Twitter connector.

8. Select the Save button on the toolbar to save your progress.

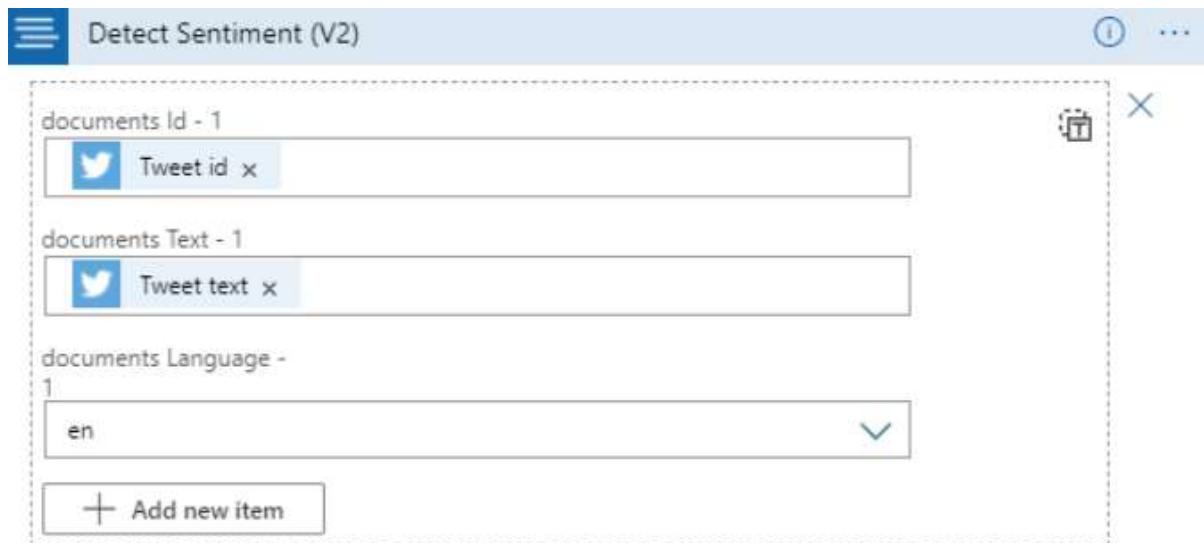
## Add Text Analytics sentiment detection

1. Select New step.
2. Search for Text Analytics in the search box.
3. Select the Text Analytics icon.
4. Select Detect Sentiment and enter the following values.

Setting	Value
Connection name	<b>TextAnalyticsConnection</b>
Account Key	Paste in the Text Analytics account key you set aside earlier.
Site URL	Paste in the Text Analytics endpoint you set aside

5. Select Create.
6. Click inside the Add new parameter box, and check the box next to documents that appears in the pop-up.
7. Click inside the documents Id - 1 textbox to open the dynamic content pop-up.
8. In the dynamic content search box, search for id, and click on Tweet id.
9. Click inside the documents Text - 1 textbox to open the dynamic content pop-up.
10. In the dynamic content search box, search for text, and click on Tweet text.
11. In Choose an action, type Text Analytics, and then click the Detect sentiment action.
12. Select the Save button on the toolbar to save your progress.

The Detect Sentiment box should look like the following screenshot.



Connected to MyCognitiveServicesConnection. [Change connection.](#)

## Connect sentiment output to function endpoint

1. Select New step.
2. Search for Azure Functions in the search box.
3. Select the Azure Functions icon.
4. Search for your function name in the search box. If you followed the guidance above, your function name begins with TweetSentimentAPI.
5. Select the function icon.
6. Select the TweetSentimentFunction item.
7. Click inside the Request Body box, and select the Detect Sentiment score item from the pop-up window.
8. Select the Save button on the toolbar to save your progress.

## Add conditional step

1. Select the **Add** an action button.
2. Click inside the Control box, and search for and select Control in the pop-up window.
3. Select Condition.
4. Click inside the Choose a value box, and select the TweetSentimentFunction Body item from the pop-up window.

5. Enter **Positive** in the Choose a value box.
6. Select the **Save** button on the toolbar to save your progress.

## Add email notifications

1. Under the True box, select the Add an action button.
2. Search for and select Office 365 Outlook in the text box.
3. Search for send and select Send an email in the text box.
4. Select the Sign in button.
5. Follow the prompts in the pop-up window to complete signing in to Office 365 Outlook.
6. Enter your email address in the To box.
7. Click inside the Subject box and click on the Body item under TweetSentimentFunction. If the Body item isn't shown in the list, click the See more link to expand the options list.
8. After the Body item in the Subject, enter the text Tweet from:.
9. After the Tweet from: text, click on the box again and select User name from the When a new tweet is posted options list.
10. Click inside the Body box and select Tweet text under the When a new tweet is posted options list. If the Tweet text item isn't shown in the list, click the See more link to expand the options list.
11. Select the Save button on the toolbar to save your progress.

The email box should now look like this screenshot.

\* Body

Font 12 **B** *I* U Tweet text x

\* Subject

Body x Tweet from: User name x

\* To

example@contoso.com

Add new parameter

Connected [Change connection.](#)

## Run the workflow

1. From your Twitter account, tweet the following text: I'm enjoying #my-twitter-tutorial.
2. Return to the Logic Apps Designer and select the Run button.
3. Check your email for a message from the workflow.

## Clean up resources

To clean up all the Azure services and accounts created during this tutorial, delete the resource group.

1. Search for Resource groups in the top search box.
2. Select the tweet-sentiment-tutorial.
3. Select Delete resource group
4. Enter tweet-sentiment-tutorial in the text box.
5. Select the Delete button.

# AZURE QUANTUM

Azure Quantum is a full-stack cloud service designed to allow users remote access to quantum computers. Azure Quantum focuses on integrating quantum computing tools and its Azure cloud service. Quantum computing focuses on making calculations based on the behaviour of particles.

## COMMANDS

<code>az quantum execute</code>	Submit a job to run on Azure Quantum, and waits for the result.
<code>az quantum job</code>	Manage jobs for Azure Quantum.
<code>az quantum job cancel</code>	Request to cancel a job on Azure Quantum if it hasn't completed.
<code>az quantum job list</code>	Get the list of jobs in a Quantum Workspace.
<code>az quantum job output</code>	Get the results of running a Q# job.
<code>az quantum job show</code>	Get the job's status and details.
<code>az quantum job submit</code>	Submit a Q# project to run on Azure Quantum.
<code>az quantum job wait</code>	Place the CLI in a waiting state until the job finishes running.
<code>az quantum offerings</code>	Manage provider offerings for Azure Quantum.
<code>az quantum offerings accept-terms</code>	Accept the terms of a provider and SKU combination to enable it for workspace creation.
<code>az quantum offerings list</code>	Get the list of all provider offerings available on the given location.
<code>az quantum offerings show-terms</code>	Show the terms of a provider and SKU combination including license URL and
<code>az quantum run</code>	Equivalent to <code>az quantum execute</code> .
<code>az quantum target</code>	Manage targets for Azure Quantum workspaces.
<code>az quantum target</code>	Clear the default target-id.
<code>az quantum target list</code>	Get the list of providers and their targets in an Azure Quantum workspace.

<code>az quantum target set</code>	Select the default target to use when submitting jobs to Azure Quantum.
<code>az quantum target show</code>	Get the details of the given (or current) target to use when submitting jobs to Azure Quantum.
<code>az quantum workspace</code>	Manage Azure Quantum workspaces.
<code>az quantum workspace clear</code>	Clear the default Azure Quantum workspace.
<code>az quantum workspace create</code>	Create a new Azure Quantum workspace.
<code>az quantum workspace delete</code>	Delete the given (or current) Azure Quantum workspace.
<code>az quantum workspace list</code>	Get the list of Azure Quantum workspaces available.
<code>az quantum workspace quotas</code>	List the quotas for the given (or current) Azure Quantum workspace.
<code>az quantum workspace set</code>	Select a default Azure Quantum workspace for future commands.
<code>az quantum workspace show</code>	Get the details of the given (or current) Azure Quantum workspace.

## **az quantum execute**

Submit a job to run on Azure Quantum, and waits for the result.

`az quantum execute [--job-name]`

`[--job-params]`

`[--location]`

`[--no-build]`

`[--project]`

`[--resource-group]`

`[--shots]`

`[--storage]`

```
[--target-id]  
[--workspace-name]  
[<PROGRAM_ARGS>]
```

## Examples

Submit the Q# program from the current folder and wait for the result.

```
az quantum execute -g MyResourceGroup -w MyWorkspace -l MyLocation -t MyTarget
```

Submit and wait for a Q# program from the current folder with job and program parameters.

```
az quantum execute -g MyResourceGroup -w MyWorkspace -l MyLocation -t MyTarget \  
--job-params key1=value1 key2=value2 -- --n-qubits=3
```

## Optional Parameters

### --job-name

A friendly name to give to this run of the program.

### --job-params

Job parameters passed to the target as a list of key=value pairs.

### --location -l

Location. Values from: az account list-locations. You can configure the default location using az configure --defaults location=<location>.

### --no-build

If specified, the Q# program is not built before submitting.

### --project

The location of the Q# project to submit. Defaults to current folder.

### --resource-group -g

Name of resource group. You can configure the default group using az configure --defaults group=<name>.

### --shots

The number of times to run the Q# program on the given target.

## **--storage**

If specified, the ConnectionString of an Azure Storage is used to store job data and results.

## **--target-id -t**

Execution engine for quantum computing jobs. When a workspace is configured with a set of provider, they each enable one or more targets. You can configure the default target using az quantum target set.

## **--workspace-name -w**

Name of the Quantum Workspace. You can configure the default workspace using az quantum workspace set.

## **<PROGRAM\_ARGS>**

List of arguments expected by the Q# operation specified as --name=value after --.

## **az quantum run**

Equivalent to az quantum execute.

az quantum run [--job-name]

[--job-params]

[--location]

[--no-build]

[--project]

[--resource-group]

[--shots]

[--storage]

[--target-id]

[--workspace-name]

[<PROGRAM\_ARGS>]

## Examples

Submit the Q# program from the current folder and wait for the result.

```
az quantum run -g MyResourceGroup -w MyWorkspace -l MyLocation -t MyTarget
```

Submit and wait for a Q# program from the current folder with job and program parameters.

```
az quantum run -g MyResourceGroup -w MyWorkspace -l MyLocation -t MyTarget \  
    --job-params key1=value1 key2=value2 -- --n-qubits=3
```

## Optional Parameters

### --job-name

A friendly name to give to this run of the program.

### --job-params

Job parameters passed to the target as a list of key=value pairs.

### --location -l

Location. Values from: az account list-locations. You can configure the default location using az configure --defaults location=<location>.

### --no-build

If specified, the Q# program is not built before submitting.

### --project

The location of the Q# project to submit. Defaults to current folder.

### --resource-group -g

Name of resource group. You can configure the default group using az configure --defaults group=<name>.

### --shots

The number of times to run the Q# program on the given target.

### --storage

If specified, the ConnectionString of an Azure Storage is used to store job data and results.

### --target-id -t

Execution engine for quantum computing jobs. When a workspace is configured with a set of provider, they each enable one or more targets. You can configure the default target using az quantum target set.

**--workspace-name -w**

Name of the Quantum Workspace. You can configure the default workspace using az quantum workspace set.

**<PROGRAM\_ARGS>**

List of arguments expected by the Q# operation specified as --name=value after --.

## **az quantum workspace**

Manage Azure Quantum workspaces.

### **Commands**

<code>az quantum workspace clear</code>	Clear the default Azure Quantum workspace.
<code>az quantum workspace create</code>	Create a new Azure Quantum workspace.
<code>az quantum workspace delete</code>	Delete the given (or current) Azure Quantum workspace.
<code>az quantum workspace list</code>	Get the list of Azure Quantum workspaces available.
<code>az quantum workspace quotas</code>	List the quotas for the given (or current) Azure Quantum workspace.
<code>az quantum workspace set</code>	Select a default Azure Quantum workspace for future commands.
<code>az quantum workspace show</code>	Get the details of the given (or current) Azure Quantum workspace.

## **az quantum workspace clear**

Clear the default Azure Quantum workspace.

```
az quantum workspace clear
```

## **az quantum workspace create**

Create a new Azure Quantum workspace.

```
az quantum workspace create [--location]
```

```
[--provider-sku-list]  
[--resource-group]  
[--skip-role-assignment]  
[--storage-account]  
[--workspace-name]
```

## **Examples**

Create a new Azure Quantum workspace with a specific list of providers.

```
az quantum workspace create -g MyResourceGroup -w MyWorkspace -l MyLocation \  
-r "MyProvider1 / MySKU1,MyProvider2 / MySKU2" -a MyStorageAccountName
```

## **Optional Parameters**

### **--location -l**

Location. Values from: az account list-locations. You can configure the default location using az configure --defaults location=<location>.

### **--provider-sku-list -r**

Comma separated list of Provider/SKU pairs.

### **--resource-group -g**

Name of resource group. You can configure the default group using az configure --defaults group=<name>.

### **--skip-role-assignment**

Skip the role assignment step for the quantum workspace in the storage account.

### **--storage-account -a**

Name of the storage account to be used by a quantum workspace.

### **--workspace-name -w**

Name of the Quantum Workspace. You can configure the default workspace using az quantum workspace set.

## **az quantum workspace delete**

Delete the given (or current) Azure Quantum workspace.

```
az quantum workspace delete [--resource-group]
```

```
    [--workspace-name]
```

### **Examples**

Delete an Azure Quantum workspace by name and group.

```
az quantum workspace delete -g MyResourceGroup -w MyWorkspace
```

Delete and clear the default Azure Quantum workspace (if one has been set).

```
az quantum workspace delete
```

## **Optional Parameters**

### **--resource-group -g**

Name of resource group. You can configure the default group using az configure --defaults group=<name>

### **--workspace-name -w**

Name of the Quantum Workspace. You can configure the default workspace using az quantum workspace set.

## az quantum workspace list

Get the list of Azure Quantum workspaces available.

```
az quantum workspace list
```

Get the list Azure Quantum workspaces available in a location

```
az quantum workspace list -l MyLocation
```

## Optional Parameters

### --location -l

Location. Values from: az account list-locations. You can configure the default location using az configure --defaults location=<location>.

### --resource-group -g

Name of resource group. You can configure the default group using az configure --defaults group=<name>.

### --tag

Show only quantum workspaces that have associated the specified tag.

## az quantum workspace quotas

List the quotas for the given (or current) Azure Quantum workspace.

```
az quantum workspace quotas [--location]
```

```
    [--resource-group]
```

```
    [--workspace-name]
```

## Examples

List the quota information of the default workspace if set.

```
az quantum workspace quotas
```

List the quota information of a specified Azure Quantum workspace.

```
az quantum workspace quotas -g MyResourceGroup -w MyWorkspace -l MyLocation
```

## Optional Parameters

### --location -l

Location. Values from: az account list-locations. You can configure the default location using az configure --defaults location=<location>.

### --resource-group -g

Name of resource group. You can configure the default group using az configure --defaults group=<name>.

### --workspace-name -w

Name of the Quantum Workspace. You can configure the default workspace using az quantum workspace set.

## az quantum workspace set

Select a default Azure Quantum workspace for future commands.

```
az quantum workspace set [--location]  
                      [--resource-group]  
                      [--workspace-name]
```

## Examples

Set the default Azure Quantum workspace.

```
az quantum workspace set -g MyResourceGroup -w MyWorkspace -l MyLocation
```

## Optional Parameters

### --location -l

Location. Values from: az account list-locations. You can configure the default location using az configure --defaults location=<location>.

### --resource-group -g

Name of resource group. You can configure the default group using az configure --defaults group=<name>.

### **--workspace-name -w**

Name of the Quantum Workspace. You can configure the default workspace using az quantum workspace set.

## **az quantum workspace show**

Get the details of the given (or current) Azure Quantum workspace.

```
az quantum workspace show [--resource-group]
```

```
[--workspace-name]
```

## **Examples**

Show the currently selected default Azure Quantum workspace.

```
az quantum workspace show
```

Show the details of a provided Azure Quantum workspace.

```
az quantum workspace show -g MyResourceGroup -w MyWorkspace
```

## **Optional Parameters**

### **--resource-group -g**

Name of resource group. You can configure the default group using az configure --defaults group=<name>.

### **--workspace-name -w**

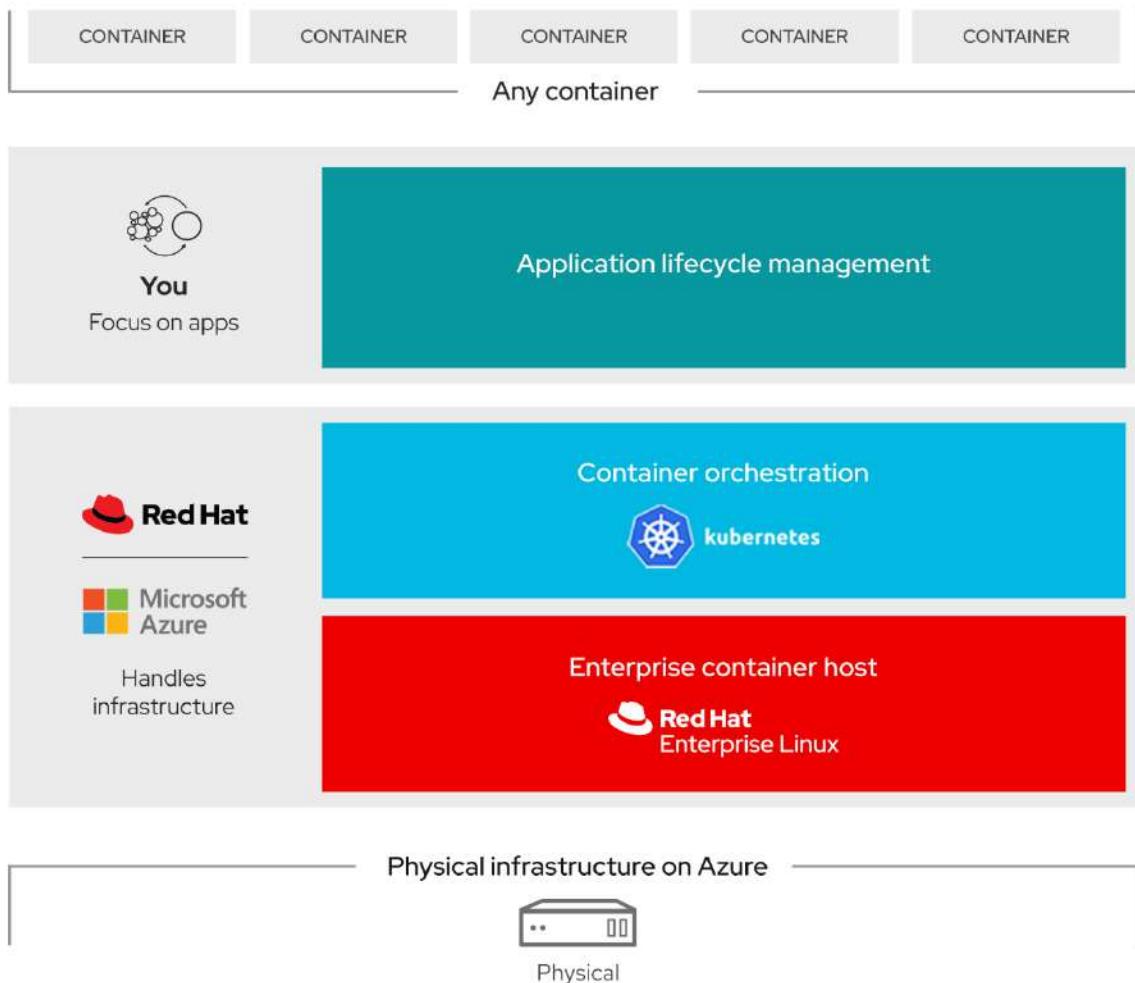
Name of the Quantum Workspace. You can configure the default workspace using az quantum workspace set.

## **AZURE RED HAT OPENSHIFT**

The Microsoft Azure Red Hat OpenShift service enables you to deploy fully managed OpenShift clusters.

Azure Red Hat OpenShift extends Kubernetes. Running containers in production with Kubernetes requires additional tools and resources. This often includes needing to juggle

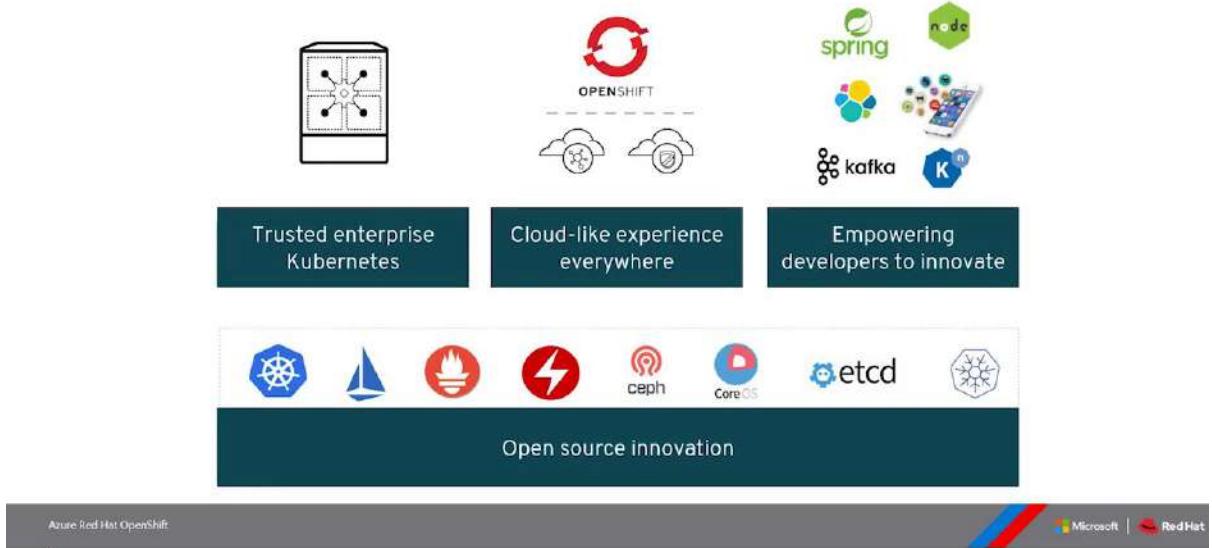
image registries, storage management, networking solutions, and logging and monitoring tools - all of which must be versioned and tested together. Building container-based applications requires even more integration work with middleware, frameworks, databases, and CI/CD tools. Azure Red Hat OpenShift combines all this into a single platform, bringing ease of operations to IT teams while giving application teams what they need to execute.



Azure Red Hat OpenShift is jointly engineered, operated, and supported by Red Hat and Microsoft to provide an integrated support experience. There are no virtual machines to operate, and no patching is required. Master, infrastructure, and application nodes are patched, updated, and monitored on your behalf by Red Hat and Microsoft. Your Azure Red Hat OpenShift clusters are deployed into your Azure subscription and are included on your Azure bill.

You can choose your own registry, networking, storage, and CI/CD solutions, or use the built-in solutions for automated source code management, container and application builds, deployments, scaling, health management, and more. Azure Red Hat OpenShift provides an integrated sign-on experience through Azure Active Directory.

## Why customers choose Red Hat OpenShift



## Access, security, and monitoring

For improved security and management, Azure Red Hat OpenShift lets you integrate with Azure Active Directory (Azure AD) and use Kubernetes role-based access control (Kubernetes RBAC). You can also monitor the health of your cluster and resources.

## Cluster and node

Azure Red Hat OpenShift nodes run on Azure virtual machines. You can connect storage to nodes and pods and upgrade cluster components.

## Service Level Agreement

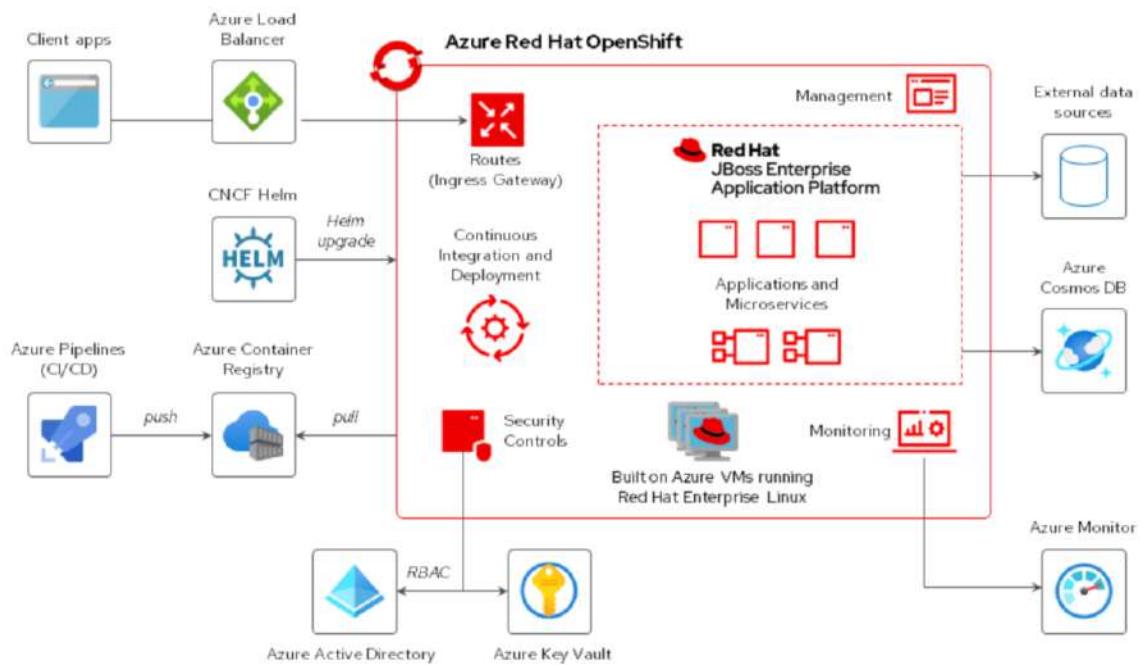
Azure Red Hat OpenShift offers a Service Level Agreement to guarantee that the service will be available 99.95% of the time.

## SUPPORT LIFECYCLE FOR AZURE RED HAT OPENSHIFT 4

Red Hat releases minor versions of Red Hat OpenShift Container Platform (OCP) roughly every three months. These releases include new features and improvements. Patch releases are more frequent (typically weekly) and are only intended for critical bug fixes.

within a minor version. These patch releases may include fixes for security vulnerabilities or major bugs.

Azure Red Hat OpenShift is built from specific releases of OCP. This article covers the versions of OCP that are supported for Azure Red Hat OpenShift and details about upgrades, deprecations, and support policy.



## Red Hat OpenShift versions

Red Hat OpenShift Container Platform uses semantic versioning. Semantic versioning uses different levels of version numbers to specify different levels of versioning. The following table illustrates the different parts of a semantic version number, in this case using the example version number 4.4.11.

Major version (x)	Minor version (y)	Patch (z)
4	4	11

Each number in the version indicates general compatibility with the previous version:

**Major version:** No major version releases are planned at this time. Major versions change when incompatible API changes or backwards compatibility may be broken.

**Minor version:** Released approximately every three months. Minor version upgrades can include feature additions, enhancements, deprecations, removals, bug fixes, security enhancements, and other improvements.

**Patches:** Typically released each week, or as needed. Patch version upgrades can include bug fixes, security enhancements, and other improvements.

Customers should aim to run the latest minor release of the major version they're running. For example, if your production cluster is on 4.4, and 4.5 is the latest generally available minor version for the 4 series, you should upgrade to 4.5 as soon as you can.

## Upgrade channels

Upgrade channels are tied to a minor version of Red Hat OpenShift Container Platform (OCP). For instance, OCP 4.4 upgrade channels will never include an upgrade to a 4.5 release. Upgrade channels control only release selection and don't impact the version of the cluster.

Azure Red Hat OpenShift 4 supports stable channels only. For example: stable-4.4.

You can use the stable-4.5 channel to upgrade from a previous minor version of Azure Red Hat OpenShift. Clusters upgraded using fast, prerelease, and candidate channels will not be supported.

If you change to a channel that does not include your current release, an alert displays and no updates can be recommended, but you can safely change back to your original channel at any point.

## Red Hat OpenShift Container Platform version support policy

Azure Red Hat OpenShift supports two generally available (GA) minor versions of Red Hat OpenShift Container Platform:

- The latest GA minor version that is released in Azure Red Hat OpenShift (which we will refer to as N)
- One previous minor version (N-1)

If available in a stable upgrade channel, newer minor releases (N+1, N+2) available in upstream OCP may be supported as well.

Critical patch updates are applied to clusters automatically by Azure Red Hat OpenShift Site Reliability Engineers (SRE). Customers that wish to install patch updates in advance are free to do so.

For example, if Azure Red Hat OpenShift introduces 4.5.z today, support is provided for the following versions:

New minor version	Supported version list
4.5.z	4.5.z, 4.4.z

".z" is representative of patch versions. If available in a stable upgrade channel, customers may also upgrade to 4.6.z.

When a new minor version is introduced, the oldest minor version is deprecated and removed. For example, say the current supported version list is 4.5.z and 4.4.z. When Azure Red Hat OpenShift releases 4.6.z, the 4.4.z release will be removed and will be out of support within 30 days.

## Release and deprecation process

For new minor versions of Red Hat OpenShift Container Platform:

- The Azure Red Hat OpenShift SRE team publishes a pre-announcement with the planned date of a new version release and respective old version deprecation on the Azure Red Hat OpenShift Release notes at least 30 days prior to removal.
- The Azure Red Hat OpenShift SRE team publishes a service health notification available to all customers with Azure Red Hat OpenShift and portal access, and sends an email to the subscription administrators with the planned version removal dates.
- Customers have 30 days from version removal to upgrade to a supported minor version release to continue receiving support.

For new patch versions of Red Hat OpenShift Container Platform:

- Because of the urgent nature of patch versions, these can be introduced into the service by Azure Red Hat OpenShift SRE team as they become available.
- In general, the Azure Red Hat OpenShift SRE team does not perform broad communications for the installation of new patch versions. However, the team constantly monitors and validates available CVE patches to support them in a timely manner. If customer action is required, the team will notify customers about the upgrade.

## **Supported versions policy exceptions**

The Azure Red Hat OpenShift SRE team reserves the right to add or remove new/existing versions, or delay upcoming minor release versions, that have been identified to have one or more critical production impacting bugs or security issues without advance notice.

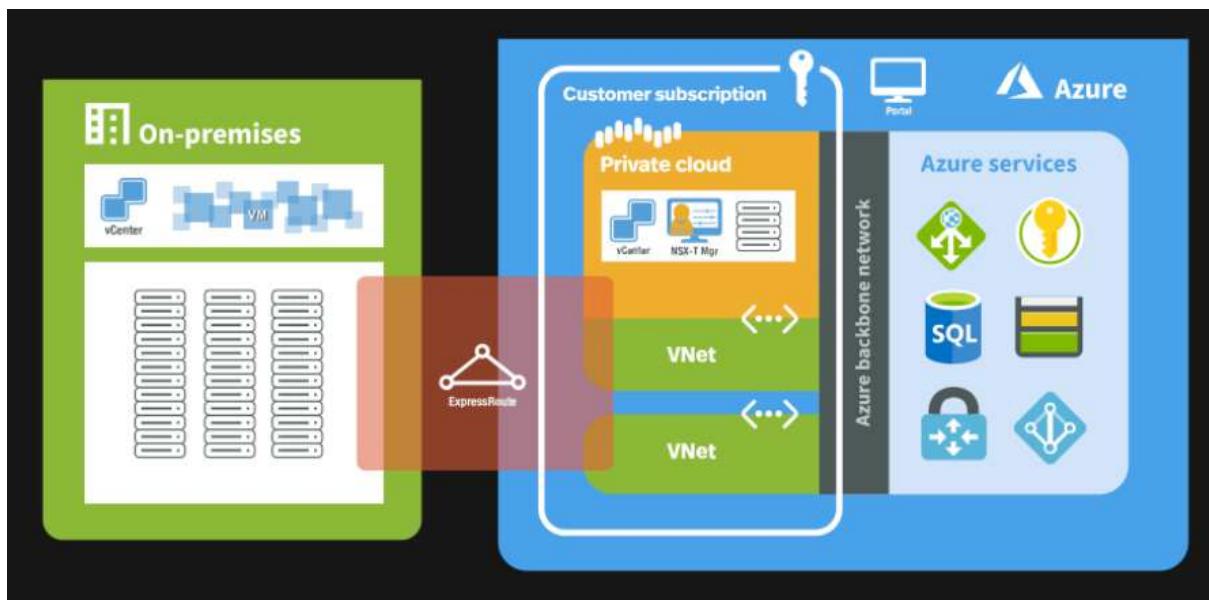
Specific patch releases may be skipped, or rollout may be accelerated depending on the severity of the bug or security issue.

## AZURE VMWARE SOLUTION

Azure VMware Solution provides you with private clouds that contain vSphere clusters built from dedicated bare-metal Azure infrastructure. The minimum initial deployment is three hosts, but additional hosts can be added one at a time, up to a maximum of 16 hosts per cluster. All provisioned private clouds have vCenter Server, vSAN, vSphere, and NSX-T. As a result, you can migrate workloads from your on-premises environments, deploy new virtual machines (VMs), and consume Azure services from your private clouds. In addition, Azure VMware Solution management tools (vCenter Server and NSX Manager) are available at least 99.9% of the time.

Azure VMware Solution is a VMware validated solution with ongoing validation and testing of enhancements and upgrades. Microsoft manages and maintains private cloud infrastructure and software. It allows you to focus on developing and running workloads in your private clouds.

The diagram shows the adjacency between private clouds and VNets in Azure, Azure services, and on-premises environments. Network access from private clouds to Azure services or VNets provides SLA-driven integration of Azure service endpoints. ExpressRoute Global Reach connects your on-premises environment to your Azure VMware Solution private cloud.



### Hosts, clusters, and private clouds

Azure VMware Solution private clouds and clusters are built from a bare-metal, hyper-converged Azure infrastructure host. The high-end (HE) hosts have 576-GB RAM and

dual Intel 18 core, 2.3-GHz processors. In addition, the HE hosts have two vSAN disk groups with 15.36 TB (SSD) of raw vSAN capacity tier and a 3.2 TB (NVMe) vSAN cache tier.

You can deploy new private clouds through the Azure portal or Azure CLI.

## Networking

Azure VMware Solution offers a private cloud environment accessible from on-premises and Azure-based resources. Services such as Azure ExpressRoute, VPN connections, or Azure Virtual WAN deliver the connectivity. However, these services require specific network address ranges and firewall ports for enabling the services.

When deploying a private cloud, private networks for management, provisioning, and vMotion get created. You'll use these private networks to access vCenter and NSX-T Manager and virtual machine vMotion or deployment.

ExpressRoute Global Reach is used to connect private clouds to on-premises environments. It connects circuits directly at the Microsoft Enterprise Edge (MSEE) level. The connection requires a virtual network (vNet) with an ExpressRoute circuit to on-premises in your subscription. The reason is that vNet gateways (ExpressRoute Gateways) can't transit traffic, which means you can attach two circuits to the same gateway, but it won't send the traffic from one circuit to the other.

Each Azure VMware Solution environment is its own ExpressRoute region (its own virtual MSEE device), which lets you connect Global Reach to the 'local' peering location. It allows you to connect multiple Azure VMware Solution instances in one region to the same peering location.

Virtual machines deployed on the private cloud are accessible to the internet through the Azure Virtual WAN public IP functionality. For new private clouds, internet access is disabled by default.

## Access and security

Azure VMware Solution private clouds use vSphere role-based access control for enhanced security. You can integrate vSphere SSO LDAP capabilities with Azure Active Directory. For more information, see the Access and Identity concepts.

vSAN data-at-rest encryption, by default, is enabled and is used to provide vSAN datastore security. For more information, see Storage concepts.

## Host and software lifecycle maintenance

Regular upgrades of the Azure VMware Solution private cloud and VMware software ensure the latest security, stability, and feature sets are running in your private clouds. For more information, see Host maintenance and lifecycle management.

## Monitoring your private cloud

Once you've deployed Azure VMware Solution into your subscription, Azure Monitor logs are generated automatically.

In your private cloud, you can:

- Collect logs on each of your VMs.
- Download and install the MMA agent on Linux and Windows VMs.
- Enable the Azure diagnostics extension.
- Create and run new queries.
- Run the same queries you usually run on your VMs.

Monitoring patterns inside the Azure VMware Solution are similar to Azure VMs within the IaaS platform. For more information and how-tos, see Monitoring Azure VMs with Azure Monitor.

## Customer communication

You can find service issues, planned maintenance, health advisories, security advisories notifications published through **Service Health** in the Azure portal. You can take timely actions when you set up activity log alerts for these notifications. You can take timely actions when you set up activity log alerts for these notifications.

Home > Service Health

## Service Health | Health advisories (2)

Search (Ctrl+ /) Subscription Region Service

**ACTIVE EVENTS**

- Service issues
- Planned maintenance (1)
- Health advisories (2)**
- Security advisories

**HISTORY**

- Health history

**RESOURCE HEALTH**

- Resource health

**ALERTS**

- Health alerts

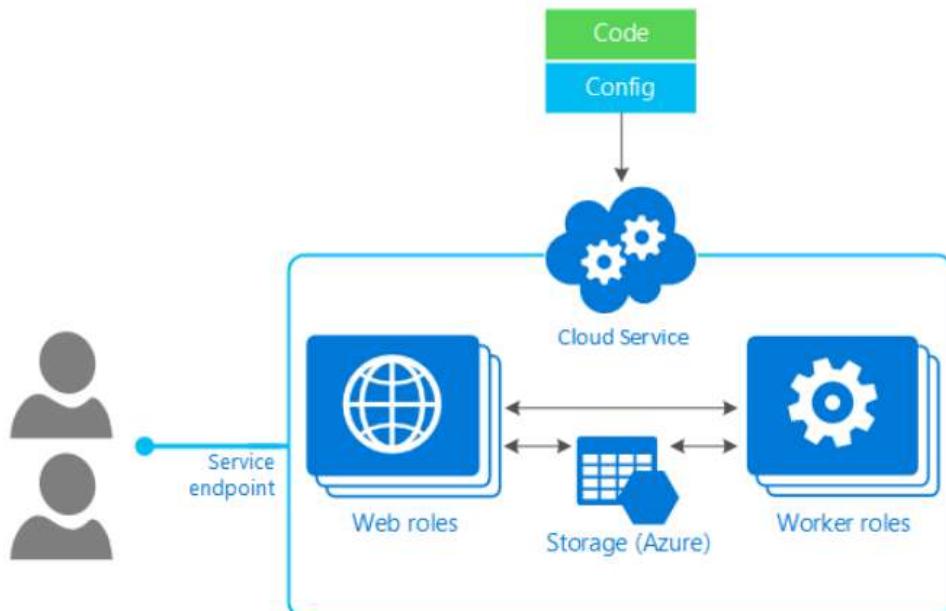
We have important information for your Azure VMware Solution service in the West US region. A private cloud instance is running low on compute and storage capacity, which can affect its performance or your Service Level Agreements. To avoid disruption to any

Create a support request

# CLOUD SERVICES

## Overview of Azure Cloud Services

Azure Cloud Services is an example of a platform as a service (PaaS). Like Azure App Service, this technology is designed to support applications that are scalable, reliable, and inexpensive to operate. In the same way that App Service is hosted on virtual machines (VMs), so too is Azure Cloud Services. However, you have more control over the VMs. You can install your own software on VMs that use Azure Cloud Services, and you can access them remotely.



More control also means less ease of use. Unless you need the additional control options, it's typically quicker and easier to get a web application up and running in the Web Apps feature of App Service compared to Azure Cloud Services.

There are two types of Azure Cloud Services roles. The only difference between the two is how your role is hosted on the VMs:

- **Web role:** Automatically deploys and hosts your app through IIS.
- **Worker role:** Does not use IIS, and runs your app standalone.

For example, a simple application might use just a single web role, serving a website. A more complex application might use a web role to handle incoming requests from users, and then pass those requests on to a worker role for processing. (This communication might use Azure Service Bus or Azure Queue storage.)

As the preceding figure suggests, all the VMs in a single application run in the same cloud service. Users access the application through a single public IP address, with requests automatically load balanced across the application's VMs. The platform scales and deploys the VMs in an Azure Cloud Services application in a way that avoids a single point of hardware failure.

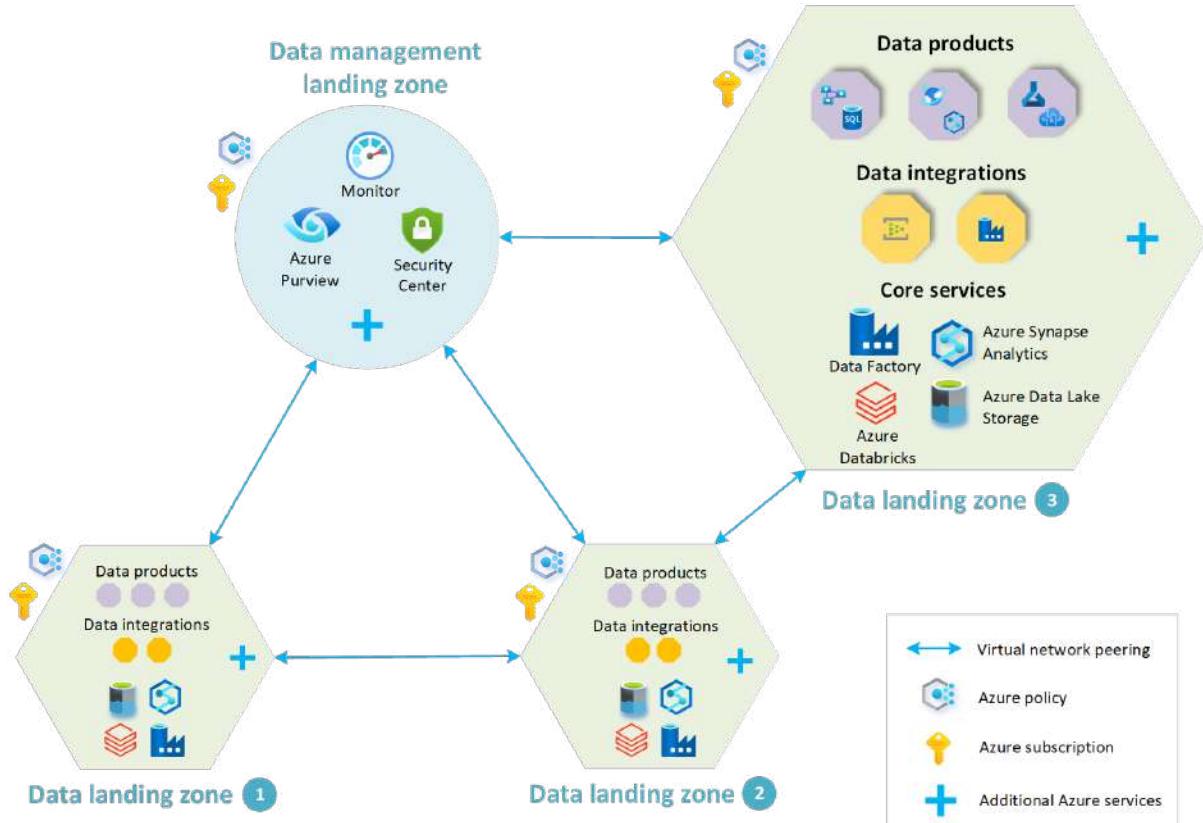
Even though applications run in VMs, it's important to understand that Azure Cloud Services provides PaaS, not infrastructure as a service (IaaS). Here's one way to think about it. With IaaS, such as Azure Virtual Machines, you first create and configure the environment your application runs in. Then you deploy your application into this environment. You're responsible for managing much of this world, by doing things such as deploying new patched versions of the operating system in each VM. In PaaS, by contrast, it's as if the environment already exists. All you have to do is deploy your application. Management of the platform it runs on, including deploying new versions of the operating system, is handled for you.

## Scaling and management

With Azure Cloud Services, you don't create virtual machines. Instead, you provide a configuration file that tells Azure how many of each you'd like, such as "three web role instances" and "two worker role instances." The platform then creates them for you. You still choose what size those backing VMs should be, but you don't explicitly create them yourself. If your application needs to handle a greater load, you can ask for more VMs, and Azure creates those instances. If the load decreases, you can shut down those instances and stop paying for them.

An Azure Cloud Services application is typically made available to users via a two-step process. A developer first uploads the application to the platform's staging area. When the developer is ready to make the application live, they use the Azure portal to swap staging

with production. This switch between staging and production can be done with no downtime, which lets a running application be upgraded to a new version without disturbing its users.



## Monitoring

Azure Cloud Services also provides monitoring. Like Virtual Machines, it detects a failed physical server and restarts the VMs that were running on that server on a new machine. But Azure Cloud Services also detects failed VMs and applications, not just hardware failures. Unlike Virtual Machines, it has an agent inside each web and worker role, and so it's able to start new VMs and application instances when failures occur.

The PaaS nature of Azure Cloud Services has other implications, too. One of the most important is that applications built on this technology should be written to run correctly when any web or worker role instance fails. To achieve this, an Azure Cloud Services application shouldn't maintain state in the file system of its own VMs. Unlike VMs created with Virtual Machines, writes made to Azure Cloud Services VMs aren't persistent. There's nothing like a Virtual Machines data disk. Instead, an Azure Cloud Services application should explicitly write all state to Azure SQL Database, blobs, tables, or some other external storage. Building applications this way makes them easier to scale and more resistant to failure, which are both important goals of Azure Cloud Services.

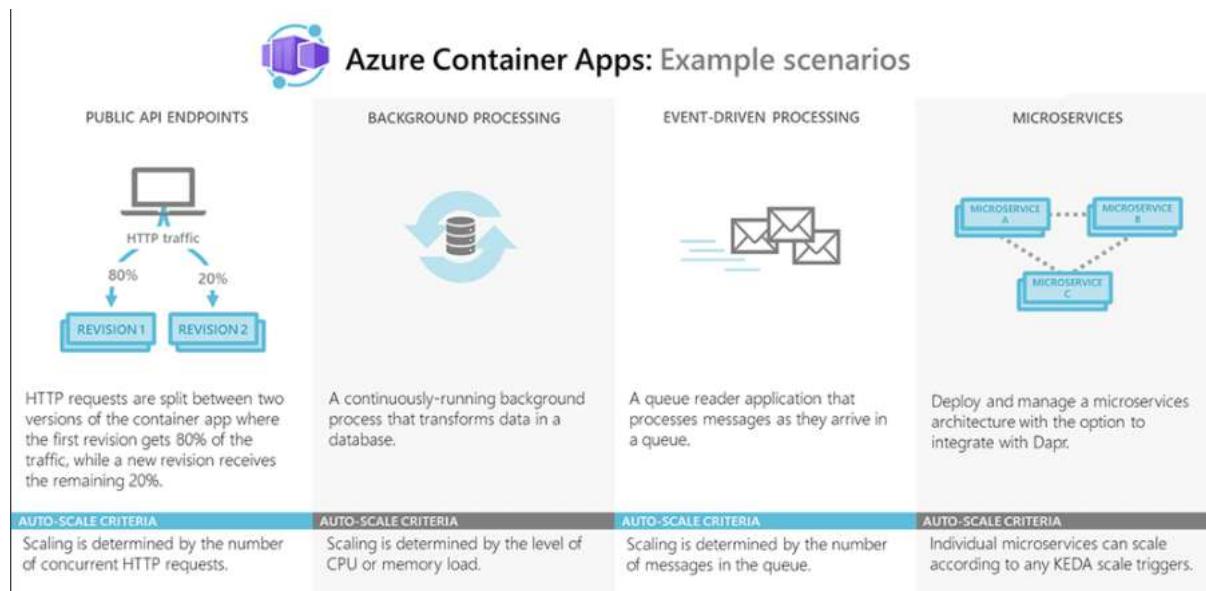
# CONTAINER APPS

Azure Container Apps enables you to run microservices and containerized applications on a serverless platform. Common uses of Azure Container Apps include:

- Deploying API endpoints
- Hosting background processing applications
- Handling event-driven processing
- Running microservices

Applications built on Azure Container Apps can dynamically scale based on the following characteristics:

- HTTP traffic
- Event-driven processing
- CPU or memory load
- Any KEDA-supported scaler



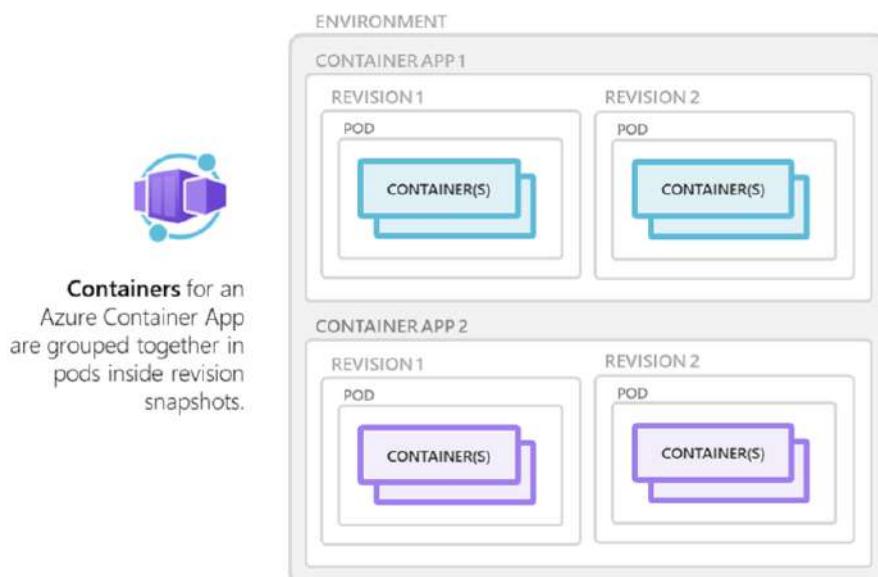
Azure Container Apps enables executing application code packaged in any container and is unopinionated about runtime or programming model. With Container Apps, you enjoy the benefits of running containers while leaving behind the concerns of managing cloud infrastructure and complex container orchestrators.

With Azure Container Apps, you can:

- **Run multiple container revisions** and manage the container app's application lifecycle.
- **Autoscale** your apps based on any KEDA-supported scale trigger. Most applications can scale to zero<sup>1</sup>.
- **Enable HTTPS ingress** without having to manage other Azure infrastructure.
- **Split traffic** across multiple versions of an application for Blue/Green deployments and A/B testing scenarios.
- **Use internal ingress and service discovery** for secure internal-only endpoints with built-in DNS-based service discovery.
- **Build microservices with Dapr** and access its rich set of APIs.
- **Run containers from any registry**, public or private, including Docker Hub and Azure Container Registry (ACR).
- **Use the Azure CLI extension or ARM templates** to manage your applications.
- **Securely manage secrets** directly in your application.
- **View application logs** using Azure Log Analytics.

Applications that scale on CPU or memory load can't scale to zero.

Azure Container Apps manages the details of Kubernetes and container orchestrations for you. Containers in Azure Container Apps can use any runtime, programming language, or development stack of your choice.



Azure Container Apps supports:

- Any Linux-based container image
  - Containers from any public or private container registry

Additional features include:

- There is no required base container image.
  - Changes to the template ARM configuration section triggers a new container app revision.
  - If a container crashes, it automatically restarts.

## Configuration

The following example configuration shows the options available when setting up a container.

```
{  
...  
"template": {  
    "containers": [  
        {  
            "image": "myacr.azurecr.io/myrepo/api-service:v1",  
            "name": "my-container-image",  
            "command": ["/bin/queue"],  
            "args": [],  
            "env": [  
                {  
                    "name": "HTTP_PORT",  
                    "value": "8080"  
                }  
            ],  
            "resources": {  
                "limits": {  
                    "cpu": "1",  
                    "memory": "1Gi"  
                },  
                "requests": {  
                    "cpu": "0.5",  
                    "memory": "512Mi"  
                }  
            }  
        }  
    ]  
},  
"build": {  
    "context": ".",  
    "dockerfile": "Dockerfile",  
    "args": {  
        "IMAGE_NAME": "my-container-image",  
        "IMAGE_TAG": "v1",  
        "IMAGE_REGISTRY": "myacr.azurecr.io/  
    }  
},  
"outputs": {  
    "apis": {  
        "api": "https://myacr.azurecr.io/v2/_catalog",  
        "username": "myuser",  
        "password": "mypassword",  
        "http": "https",  
        "port": 443  
    }  
},  
"resources": {  
    "limits": {  
        "cpu": "1",  
        "memory": "1Gi"  
    },  
    "requests": {  
        "cpu": "0.5",  
        "memory": "512Mi"  
    }  
},  
"variables": {}  
}
```

```

    "cpu": 1,
    "memory": "250Mb"
  }
}

}

```

<b>Setting</b>	<b>Description</b>	<b>Remarks</b>
image	The container image name for your container app.	This value takes the form of <code>repository/image-name:tag</code> .
name	Friendly name of the container.	Used for reporting and identification.
command	The container's startup command.	Equivalent to Docker's <a href="#">entrypoint</a> field.
args	Start up command arguments.	Entries in the array are joined together to create a parameter list to pass to the startup command.
env	An array of key/value pairs that define environment variables.	Use <code>secretRef</code> instead of the <code>value</code> field to refer to a secret.
resources.cpu	The number of CPUs allocated to the container.	Values must adhere to the following rules: the value must be greater than zero and less than 2, and can be any decimal number, with a maximum of one decimal place. For example, <code>1.1</code> is valid, but <code>1.55</code> is invalid. The default is 0.5 CPU per container.

<b>Setting</b>	<b>Description</b>	<b>Remarks</b>
<code>resources.memory</code>	The amount of RAM allocated to the container.	This value is up to <code>4Gi</code> . The only allowed units are <a href="#">gibibytes</a> ( <code>Gi</code> ). Values must adhere to the following rules: the value must be greater than zero and less than <code>4Gi</code> , and can be any decimal number, with a maximum of two decimal places. For example, <code>1.25Gi</code> is valid, but <code>1.555Gi</code> is invalid. The default is <code>1Gi</code> per container.

The total amount of CPUs and memory requested for all the containers in a container app must add up to one of the following combinations.

<b>vCPUs</b>	<b>Memory in Gi</b>
0.5	1.0
1.0	2.0
1.5	3.0
2.0	4.0

- All of the CPU requests in all of your containers must match one of the values in the vCPUs column.
- All of the memory requests in all your containers must match the memory value in the memory column in the same row of the CPU column.

## Multiple containers

You can define multiple containers in a single container app. Groups of containers are known as pods. The containers in a pod share hard disk and network resources and experience the same application lifecycle.

You run multiple containers together by defining more than one container in the configuration's containers array.

Reasons to run containers together in a pod include:

- Use a container as a sidecar to your primary app.
- Use of a shared disk space and virtual network.

- Share scale rules among containers.
- Group together multiple containers that need to always run together.
- Enable direct communication among containers on the same host.

## Container registries

You can deploy images hosted on private registries where credentials are provided through the Container Apps configuration.

To use a container registry, you first define the required fields to the configuration's registries section.

```
{
  ...
  "registries": {
    "server": "docker.io",
    "username": "my-registry-user-name",
    "passwordSecretRef": "my-password-secretfref-name"
  }
}
```

With this set up, the saved credentials can be used when you reference a container image in an image in the containers array.

The following example shows how to deploy an app from the Azure Container Registry.

```
{
  ...
  "configuration": {
    "secrets": [
      {
        "name": "acr-password",
        "value": "my-acr-password"
      }
    ]
  }
}
```

```
],
  "registries": [
    {
      "server": "myacr.azurecr.io",
      "username": "someuser",
      "passwordSecretRef": "acr-password"
    }
  ]
}
```

## Limitations

Azure Container Apps has the following limitations:

- **Privileged containers:** Azure Container Apps can't run privileged containers. If your program attempts to run a process that requires root access, the application inside the container experiences a runtime error.
- **Operating system:** Linux-based container images are required.

## Deploy your first container app

Azure Container Apps Preview enables you to run microservices and containerized applications on a serverless platform. With Container Apps, you enjoy the benefits of running containers while leaving behind the concerns of manually configuring cloud infrastructure and complex container orchestrators.

## Setup

Begin by signing in to Azure from the CLI. Run the following command, and follow the prompts to complete the authentication process.

```
az login
```

Next, install the Azure Container Apps extension to the CLI.

```
az extension add \
--source https://workerappscliextension.blob.core.windows.net/azure-cli-extension/containerapp-0.2.0-
py2.py3-none-any.whl
```

Now that the extension is installed, register the Microsoft.Web namespace.

```
az provider register --namespace Microsoft.Web
```

Next, set the following environment variables:

```
RESOURCE_GROUP="my-container-apps"
LOCATION="canadacentral"
LOG_ANALYTICS_WORKSPACE="my-container-apps-logs"
CONTAINERAPPS_ENVIRONMENT="my-environment"
```

With these variables defined, you can create a resource group to organize the services related to your new container app.

```
az group create \
--name $RESOURCE_GROUP \
--location "$LOCATION"
```

With the CLI upgraded and a new resource group available, you can create a Container Apps environment and deploy your container app.

## Create an environment

An environment in Azure Container Apps creates a secure boundary around a group of container apps. Container Apps deployed to the same environment are deployed in the same virtual network and write logs to the same Log Analytics workspace.

Azure Log Analytics is used to monitor your container app required when creating a Container Apps environment.

Create a new Log Analytics workspace with the following command:

```
az monitor log-analytics workspace create \
--resource-group $RESOURCE_GROUP \
--workspace-name $LOG_ANALYTICS_WORKSPACE
```

Next, retrieve the Log Analytics Client ID and client secret.

```
LOG_ANALYTICS_WORKSPACE_CLIENT_ID=$(az monitor log-analytics workspace show --query
customerId -g $RESOURCE_GROUP -n $LOG_ANALYTICS_WORKSPACE -o tsv | tr -d '[:space:]')
```

```
LOG_ANALYTICS_WORKSPACE_CLIENT_SECRET=`az monitor log-analytics workspace get-shared-keys --query primarySharedKey -g $RESOURCE_GROUP -n $LOG_ANALYTICS_WORKSPACE -o tsv | tr -d ':space:'`
```

The Log Analytics values are used as you create the Container Apps environment.

To create the environment, run the following command:

```
az containerapp env create \
--name $CONTAINERAPPS_ENVIRONMENT \
--resource-group $RESOURCE_GROUP \
--logs-workspace-id $LOG_ANALYTICS_WORKSPACE_CLIENT_ID \
--logs-workspace-key $LOG_ANALYTICS_WORKSPACE_CLIENT_SECRET \
--location "$LOCATION"
```

## Create a container app

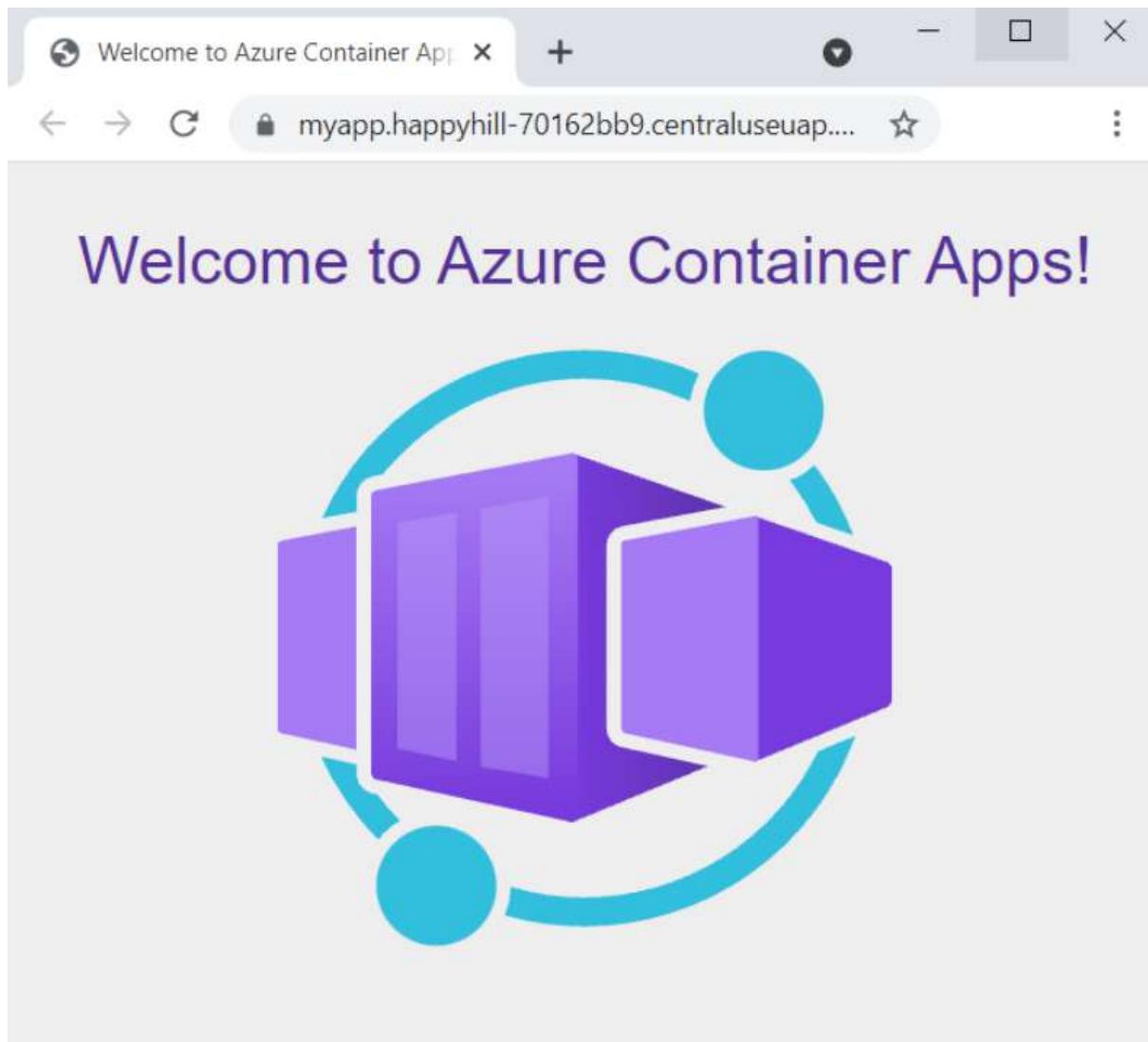
Now that you have an environment created, you can deploy your first container app. Using the containerapp create command, deploy a container image to Azure Container Apps.

```
az containerapp create \
--name my-container-app \
--resource-group $RESOURCE_GROUP \
--environment $CONTAINERAPPS_ENVIRONMENT \
--image mcr.microsoft.com/azuredocs/containerapps-helloworld:latest \
--target-port 80 \
--ingress 'external' \
--query configuration.ingress.fqdn
```

By setting --ingress to external, you make the container app available to public requests.

## Verify deployment

The create command returned the container app's fully qualified domain name. Copy this location to a web browser and you'll see the following message.



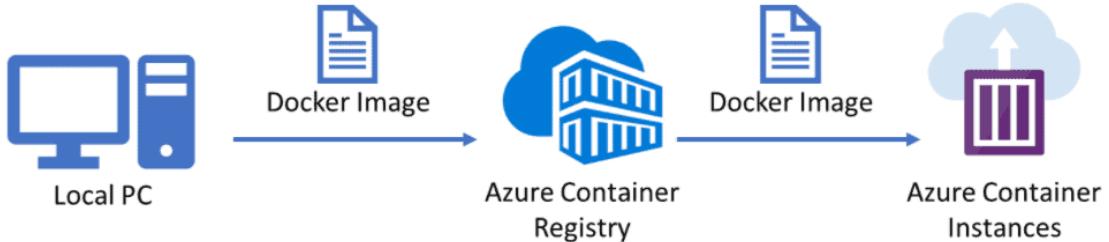
## Clean up resources

If you're not going to continue to use this application, you can delete the Azure Container Apps instance and all the associated services by removing the resource group.

```
az group delete \
--name $RESOURCE_GROUP
```

# CONTAINER INSTANCES

Containers are becoming the preferred way to package, deploy, and manage cloud applications. Azure Container Instances offers the fastest and simplest way to run a container in Azure, without having to manage any virtual machines and without having to adopt a higher-level service.



Azure Container Instances is a great solution for any scenario that can operate in isolated containers, including simple applications, task automation, and build jobs. For scenarios where you need full container orchestration, including service discovery across multiple containers, automatic scaling, and coordinated application upgrades, Azure Kubernetes Service (AKS) would be helpful.

## Fast startup times

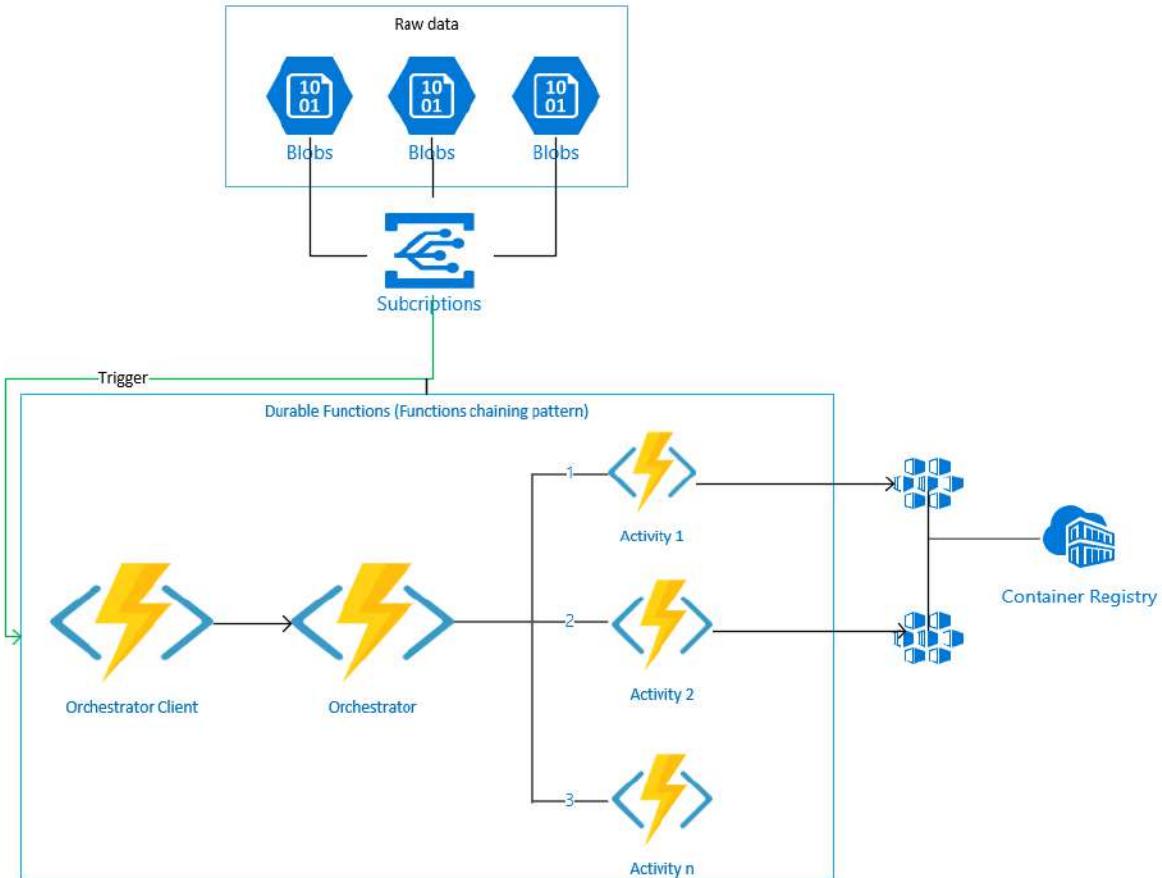
Containers offer significant startup benefits over virtual machines (VMs). Azure Container Instances can start containers in Azure in seconds, without the need to provision and manage VMs.

Bring Linux or Windows container images from Docker Hub, a private Azure container registry, or another cloud-based docker registry. Visit the FAQ to learn which registries are supported by ACI. Azure Container Instances caches several common base OS images, helping speed deployment of your custom application images.

## Container access

Azure Container Instances enables exposing your container groups directly to the internet with an IP address and a fully qualified domain name (FQDN). When you create a container instance, you can specify a custom DNS name label so your application is reachable at *customlabel.azureregion.azurecontainer.io*.

Azure Container Instances also supports executing a command in a running container by providing an interactive shell to help with application development and troubleshooting. Access takes place over HTTPS, using TLS to secure client connections.



## Compliant deployments

### Hypervisor-level security

Historically, containers have offered application dependency isolation and resource governance but have not been considered sufficiently hardened for hostile multi-tenant usage. Azure Container Instances guarantees your application is as isolated in a container as it would be in a VM.

### Customer data

The ACI service stores the minimum customer data required to ensure your container groups are running as expected. Storing customer data in a single region is currently only available in the Southeast Asia Region (Singapore) of the Asia Pacific Geo and Brazil South (Sao Paulo State) Region of Brazil Geo. For all other regions, customer data is stored in Geo. Please get in touch with Azure Support to learn more.

## Custom sizes

Containers are typically optimized to run just a single application, but the exact needs of those applications can differ greatly. Azure Container Instances provides optimum utilization by allowing exact specifications of CPU cores and memory. You pay based on what you need and get billed by the second, so you can fine-tune your spending based on actual need.

For compute-intensive jobs such as machine learning, Azure Container Instances can schedule Linux containers to use NVIDIA Tesla GPU resources (preview).

## Persistent storage

To retrieve and persist state with Azure Container Instances, we offer direct mounting of Azure Files shares backed by Azure Storage.

## Linux and Windows containers

Azure Container Instances can schedule both Windows and Linux containers with the same API. Simply specify the OS type when you create your container groups.

Some features are currently restricted to Linux containers:

- Multiple containers per container group
- Volume mounting (Azure Files, emptyDir, GitRepo, secret)
- Resource usage metrics with Azure Monitor
- Virtual network deployment
- GPU resources (preview)

For Windows container deployments, use images based on common Windows base images.

## Co-scheduled groups

Azure Container Instances supports scheduling of multi-container groups that share a host machine, local network, storage, and lifecycle. This enables you to combine your main application container with other supporting role containers, such as logging sidecars.

## **Virtual network deployment**

Azure Container Instances enables deployment of container instances into an Azure virtual network. When deployed into a subnet within your virtual network, container instances can communicate securely with other resources in the virtual network, including those that are on premises (through VPN gateway or ExpressRoute).

## **CONTAINER REGISTRY**

Azure Container Registry allows you to build, store, and manage container images and artifacts in a private registry for all types of container deployments. Use Azure container registries with your existing container development and deployment pipelines. Use Azure Container Registry Tasks to build container images in Azure on-demand, or automate builds triggered by source code updates, updates to a container's base image, or timers.

Azure Container Registry is a managed, private Docker registry service based on the open-source Docker Registry 2.0. Create and maintain Azure container registries to store and manage your private Docker container images and related artifacts.

Use Azure container registries with your existing container development and deployment pipelines, or use Azure Container Registry Tasks to build container images in Azure. Build on demand, or fully automate builds with triggers such as source code commits and base image updates.

### **Use cases**

Pull images from an Azure container registry to various deployment targets:

- **Scalable orchestration systems** that manage containerized applications across clusters of hosts, including Kubernetes, DC/OS, and Docker Swarm.
- **Azure services** that support building and running applications at scale, including Azure Kubernetes Service (AKS), App Service, Batch, Service Fabric, and others.

Developers can also push to a container registry as part of a container development workflow. For example, target a container registry from a continuous integration and delivery tool such as Azure Pipelines or Jenkins.

Configure ACR Tasks to automatically rebuild application images when their base images are updated, or automate image builds when your team commits code to a Git repository. Create multi-step tasks to automate building, testing, and patching multiple container images in parallel in the cloud.

Azure provides tooling including the Azure CLI, the Azure portal, and API support to manage your Azure container registries. Optionally install the Docker Extension for Visual Studio Code and the Azure Account extension to work with your Azure container registries. Pull and push images to an Azure container registry, or run ACR Tasks, all within Visual Studio Code.

## Key features

- **Registry service tiers** - Create one or more container registries in your Azure subscription. Registries are available in three tiers: Basic, Standard, and Premium, each of which supports webhook integration, registry authentication with Azure Active Directory, and delete functionality. Take advantage of local, network-close storage of your container images by creating a registry in the same Azure location as your deployments. Use the geo-replication feature of Premium registries for advanced replication and container image distribution scenarios.
- **Security and access** - You log in to a registry using the Azure CLI or the standard docker login command. Azure Container Registry transfers container images over HTTPS, and supports TLS to secure client connections.

You control access to a container registry using an Azure identity, an Azure Active Directory-backed service principal, or a provided admin account. Use Azure role-based access control (Azure RBAC) to assign users or systems fine-grained permissions to a registry.

Security features of the Premium service tier include content trust for image tag signing, and firewalls and virtual networks (preview) to restrict access to the registry. Microsoft Defender for Cloud optionally integrates with Azure Container Registry to scan images whenever an image is pushed to a registry.

- **Supported images and artifacts** - Grouped in a repository, each image is a read-only snapshot of a Docker-compatible container. Azure container registries can include both Windows and Linux images. You control image names for all your container deployments. Use standard Docker commands to push images into a repository, or pull an image from a repository. In addition to Docker container images, Azure Container Registry stores related content formats such as Helm charts and images built to the Open Container Initiative (OCI) Image Format Specification.

- **Automated image builds** - Use Azure Container Registry Tasks (ACR Tasks) to streamline building, testing, pushing, and deploying images in Azure. For example, use ACR Tasks to extend your development inner-loop to the cloud by offloading docker build operations to Azure. Configure build tasks to automate your container OS and framework patching pipeline, and build images automatically when your team commits code to source control.

Multi-step tasks provide step-based task definition and execution for building, testing, and patching container images in the cloud. Task steps define individual container image build and push operations. They can also define the execution of one or more containers, with each step using the container as its execution environment.

## Create an Azure container registry using the Azure portal

Azure Container Registry is a private registry service for building, storing, and managing container images and related artifacts. In this quickstart, you create an Azure container registry instance with the Azure portal. Then, use Docker commands to push a container image into the registry, and finally pull and run the image from your registry.

To log in to the registry to work with container images, this quickstart requires that you are running the Azure CLI (version 2.0.55 or later recommended). Run `az --version` to find the version. If you need to install or upgrade, see [Install Azure CLI](#).

You must also have Docker installed locally. Docker provides packages that easily configure Docker on any Mac, Windows, or Linux system.

### Sign in to Azure

Sign in to the Azure portal at <https://portal.azure.com>.

### Create a container registry

Select **Create a resource > Containers > Container Registry**.

The screenshot shows the Microsoft Azure 'New' blade. At the top, there's a search bar and a user profile icon. Below the search bar, there's a 'Search the Marketplace' input field. The main area has two tabs: 'Azure Marketplace' (selected) and 'Featured'. Under 'Azure Marketplace', there are several categories: 'Get started', 'Recently created', 'AI + Machine Learning', 'Analytics', 'Blockchain', 'Compute', 'Containers' (which is highlighted with a red dashed box), 'Databases', and 'Developer Tools'. Under 'Featured', there are four items: 'Container Instances' (with a trash bin icon), 'Container Registry' (with a cloud icon, also highlighted with a red box), 'Kubernetes Service' (with a cluster icon), and 'Web App for Containers' (with a globe icon). A small note at the bottom right says 'DC/OS on Azure (preview)'.

In the **Basics** tab, enter values for **Resource group** and **Registry name**. The registry name must be unique within Azure, and contain 5-50 alphanumeric characters. For this quickstart create a new resource group in the West US location named myResourceGroup, and for **SKU**, select 'Basic'.

The screenshot shows the 'Create container registry' page in the Azure portal. The top navigation bar includes 'Home > Container registries >' followed by the title 'Create container registry'. Below the title is a 'Create' button with a cloud icon. The page has a 'Basics' tab selected, along with 'Networking', 'Encryption', 'Tags', and 'Review + create' tabs. A descriptive text block explains that Azure Container Registry allows building, storing, and managing container images and artifacts in a private registry for all types of container deployments. It mentions using Azure Container Registry Tasks to build container images on-demand or automate builds triggered by source code updates. A 'Learn more' link is provided. The 'Project details' section includes fields for 'Subscription' (set to 'Visual Studio Enterprise Subscription') and 'Resource group' (set to '(New) myresourcegroup' with a 'Create new' link). The 'Instance details' section includes fields for 'Registry name' (set to 'mycontainerregistry'), 'Location' (set to 'West US'), and 'SKU' (set to 'Basic'). At the bottom, there are 'Review + create' and 'Next: Networking >' buttons.

Accept default values for the remaining settings. Then select **Review + create**. After reviewing the settings, select **Create**.

When the **Deployment succeeded** message appears, select the container registry in the portal.

Home >

## mycontainerregistry

Container registry

Search (Cmd+ /)

Move Delete Update

**Overview**

Activity log

Access control (IAM)

Tags

Quick start

Events

Resource group (change)  
myresourcegroup

Location  
West US

Subscription (change)  
Visual Studio Enterprise Subscription

Subscription ID

Login server  
mycontainerregistry.azurecr.io

Creation date  
8/4/2020, 5:04 PM PDT

SKU  
Basic

Provisioning state  
Succeeded

**Usage**

Included in SKU	Used	Additional stor...
<b>10.0 GiB</b>	<b>0.00 GiB</b>	<b>0.00 GiB</b>

**ACR Tasks**

Build, Run, Push and Patch containers in Azure with ACR Tasks. Tasks supports Windows, Linux and ARM with QEMU.

Learn more

Take note of the registry name and the value of the **Login server**, which is a fully qualified name ending with `azurecr.io` in the Azure cloud. You use these values in the following steps when you push and pull images with Docker.

## Log in to registry

Before pushing and pulling container images, you must log in to the registry instance. Sign into the Azure CLI on your local machine, then run the `az acr login` command. Specify only the registry resource name when logging in with the Azure CLI. Don't use the fully qualified login server name.

```
az acr login --name <registry-name>
```

Example:

```
az acr login --name mycontainerregistry
```

The command returns Login Succeeded once completed.

## Push image to registry

To push an image to an Azure Container registry, you must first have an image. If you don't yet have any local container images, run the following docker pull command to pull an existing public image. For this example, pull the hello-world image from Microsoft Container Registry.

```
docker pull mcr.microsoft.com/hello-world
```

Before you can push an image to your registry, you must tag it with the fully qualified name of your registry login server. The login server name is in the format *<registry-name>.azurecr.io* (must be all lowercase), for example, *mycontainerregistry.azurecr.io*.

Tag the image using the docker tag command. Replace *<login-server>* with the login server name of your ACR instance.

```
docker tag mcr.microsoft.com/hello-world <login-server>/hello-world:v1
```

Example:

```
docker tag mcr.microsoft.com/hello-world mycontainerregistry.azurecr.io/hello-world:v1
```

Finally, use docker push to push the image to the registry instance. Replace *<login-server>* with the login server name of your registry instance. This example creates the **hello-world** repository, containing the hello-world:v1 image.

```
docker push <login-server>/hello-world:v1
```

After pushing the image to your container registry, remove the hello-world:v1 image from your local Docker environment. (Note that this docker rmi command does not remove the image from the **hello-world** repository in your Azure container registry.)

```
docker rmi <login-server>/hello-world:v1
```

## List container images

To list the images in your registry, navigate to your registry in the portal and select **Repositories**, then select the **hello-world** repository you created with docker push.

The screenshot shows the Azure Container Registry interface for the 'mycontainerregistry' account. On the left, there's a sidebar with various navigation links: Overview, Activity log, Access control (IAM), Tags, Quick start, Events, Settings (with options like Access keys, Encryption, Identity, Networking, Security, Locks, and Export template), and Services (with options like Repositories, Webhooks, and Replications). The 'Repositories' link under Services is highlighted with a red box. The main content area has a search bar at the top labeled 'Search to filter repositories ...'. Below it, a list of repositories is shown, with 'hello-world' also highlighted with a red box. There's a refresh button and a three-dot menu icon on the right.

By selecting the **hello-world** repository, you see the v1-tagged image under **Tags**.

## Run image from registry

Now, you can pull and run the hello-world:v1 container image from your container registry by using docker run:

```
docker run <login-server>/hello-world:v1
```

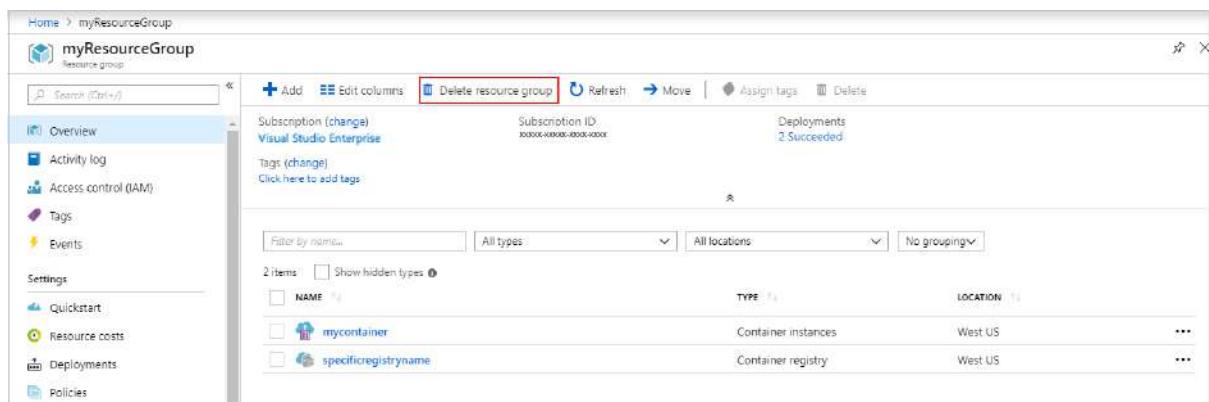
Example output:

```
Unable to find image 'mycontainerregistry.azurecr.io/hello-world:v1' locally
v1: Pulling from hello-world
Digest: sha256:662dd8e65ef7ccf13f417962c2f77567d3b132f12c95909de6c85ac3c326a345
```

Status: Downloaded newer image for mycontainerregistry.azurecr.io/hello-world:v1  
Hello from Docker!  
This message shows that your installation appears to be working correctly.  
[...]

## Clean up resources

To clean up your resources, navigate to the **myResourceGroup** resource group in the portal. Once the resource group is loaded, click on **Delete resource group** to remove the resource group, the container registry, and the container images stored there.



The screenshot shows the Azure Resource Group overview page for 'myResourceGroup'. The left sidebar includes links for Overview, Activity log, Access-control (IAM), Tags, Events, Settings, Quickstart, Resource costs, Deployments, and Policies. The main area displays resource details: Subscription (Visual Studio Enterprise), Subscription ID (10000-0000-0000-0000), and Deployments (2 Succeeded). A table lists resources: 'mycontainer' (Container instances, West US) and 'specificregistryname' (Container registry, West US). At the top, there are buttons for Add, Edit columns, Delete resource group (highlighted in red), Refresh, Move, Assign tags, and Delete.

## CYCLECLOUD

Azure CycleCloud is designed to enable enterprise IT organizations to provide secure and flexible cloud HPC and Big Compute environments to their end users. With dynamic scaling of clusters, the business can get the resources it needs at the right time and the right price. Azure CycleCloud's automated configuration enables IT to focus on providing service to the business users.

Azure CycleCloud is an enterprise-friendly tool for orchestrating and managing High Performance Computing (HPC) environments on Azure. With CycleCloud, users can provision infrastructure for HPC systems, deploy familiar HPC schedulers, and automatically scale the infrastructure to run jobs efficiently at any scale. Through CycleCloud, users can create different types of file systems and mount them to the compute cluster nodes to support HPC workloads.

Azure CycleCloud is targeted at HPC administrators and users who want to deploy an HPC environment with a specific scheduler in mind -- commonly used schedulers such as

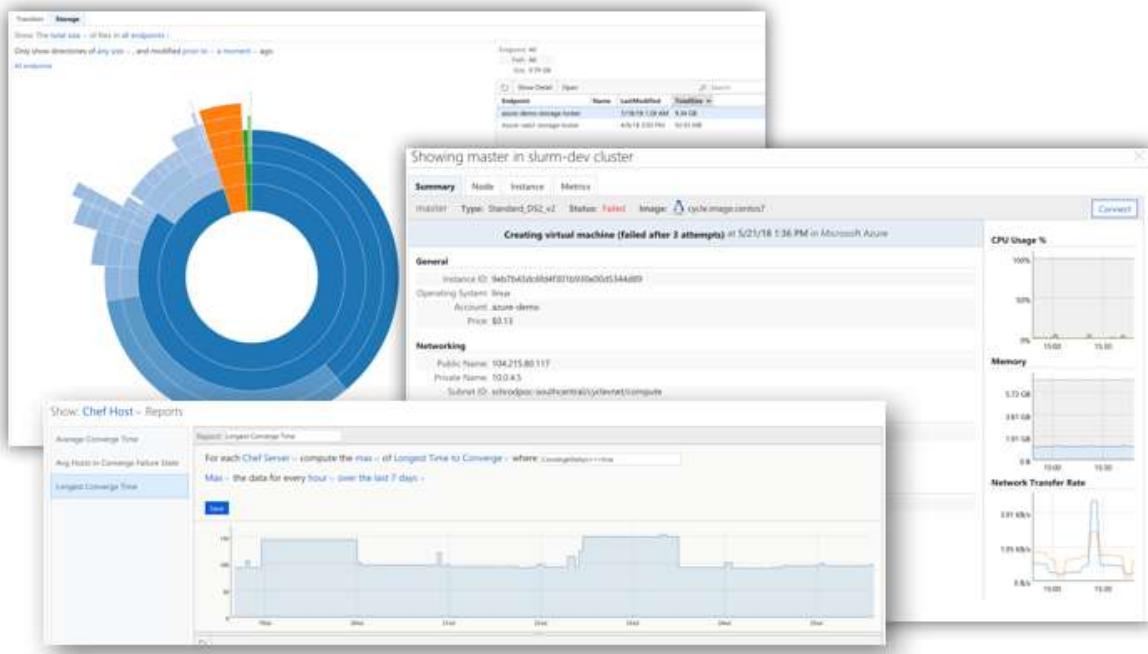
Slurm, PBSPro, LSF, Grid Engine, and HT-Condor are supported out of the box. CycleCloud is the sister product to Azure Batch, which provides a Scheduler as a Service on Azure.

CycleCloud abstracts away the basic Azure building blocks such as VMs, scalesets, network interfaces, and disks. This allows an HPC administrator to focus on the familiar: an HPC cluster comprising of nodes and a configurable scheduler of choice.

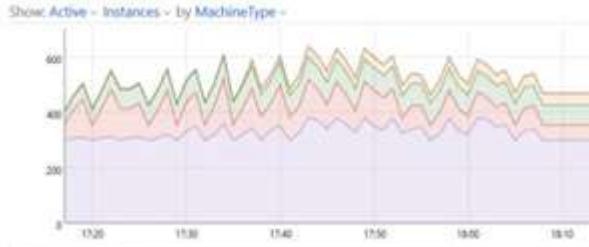
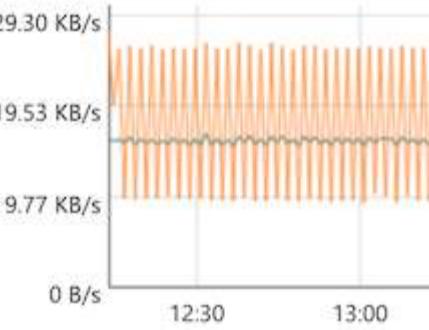
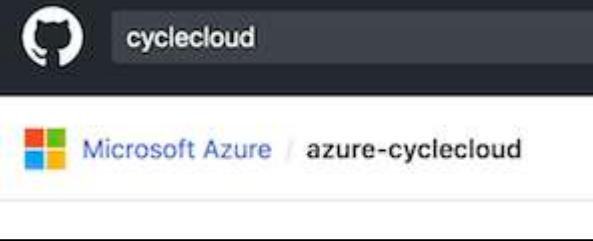
CycleCloud deploys autoscaling plugins on top of the supported schedulers, so users do not need to implement complex autoscaling functions and routines themselves, but rather interface only with scheduler-level configurations that they are familiar with.

With a rich, declarative, templating format, CycleCloud provides powerful tooling to construct complete HPC environments on Azure. Users can deploy environments that include NFS servers, parallel file systems, login hosts, license servers, and directory services -- essentially all the components needed in an HPC system -- through a single management plane.

CycleCloud integrates with Azure services such as Azure Monitor and Azure Cost Management tools.



## CycleCloud Capabilities

<p><b>Scheduler Agnostic</b></p> <p>Use standard HPC schedulers such as Slurm, PBS Pro, LSF, Grid Engine, and HTCondor, or extend CycleCloud autoscaling plugins to work with your own scheduler</p>	<p>Create a New Cluster</p> <p>Schedulers</p>  <p>TACC OpenPBS Slurm UNIVA</p>
<p><b>Manage Compute Resources</b></p> <p>Manage virtual machines and scale sets to provide a flexible set of compute resources that can meet your dynamic workload requirements</p>	<p>Compute Configurations</p> <p>Configure the execute array for the cluster. VM sizes and autoscaling limits.</p> <p>Compute Type: Standard_D2s_v3 <input type="button" value="Choose"/></p> <p>GPU Type: Standard_NC12 <input type="button" value="Choose"/></p> <p>Autoscale: <input checked="" type="checkbox"/> Start and stop execute nodes automatically</p> <p>Max Cores: <input type="text" value="500"/></p>
<p><b>Autoscale Resources</b></p> <p>Automatically adjust cluster size and components based upon job load, availability, and time requirements</p>	
<p><b>Monitor and Analyze</b></p> <p>Collect and analyze performance data using visualization tools</p>	<p>Network Transfer Rate</p>  <p>29.30 KB/s 19.53 KB/s 9.77 KB/s 0 B/s</p> <p>12:30 13:00</p>
<p><b>Template Clusters</b></p> <p>Use CycleCloud templates to share cluster topologies with the community</p>	 <p>cyclecloud</p> <p>Microsoft Azure / azure-cyclecloud</p>

<p><b><a href="#">Customize and Extend</a></b></p> <p><b><a href="#">Functionality</a></b></p> <p>Use the comprehensive RESTful API to customize and extend functionality, deploy your own scheduler, and support into existing workload managers</p>	<p><code>POST /clusters/{cluster}/nodes/remove</code></p> <p><b>Description</b></p> <p>This operation removes nodes in a cluster. The nodes can be identified by filter. Note that by default nodes are removed when terminated (i.e., when they have no active jobs).</p>
<p><b><a href="#">Integrate into Existing Workflows</a></b></p> <p>Integrate into existing workflows and tools using the built-in CLI</p>	<pre>Usage: cyclecloud create_cluster TEMPLATE_NAME [options] Creates a cluster from an existing template.  Options:   -h, --help           show this help message and exit   -v, --version        Shows the version for the CLI.   --config=CONFIG_FILE Specifies the path to a non-default configuration file to be used for this command.</pre>

## How Do I Use Azure CycleCloud?

Azure CycleCloud is an installable web application that you can run on premise or in an Azure VM. Once installed, CycleCloud can be configured to use compute and data resources in your prepared Azure subscription. CycleCloud provides a number of official cluster templates for schedulers (PBSPro, LSF, Grid Engine, Slurm, HTCondor), and filesystems (NFS, BeeGFS). Cluster templates provided by the CycleCloud community are also available. You can use these cluster templates unmodified or you can customize them for your specific needs.

Once a cluster is created, it is automatically configured to autoscale by default to handle the computational jobs that are submitted to the scheduler. CycleCloud administrative features govern access to the CycleCloud cluster for other users in your organization.

Tooling using templates and configuration scripts enable you to build complex HPC environments quickly, and replicate these for separate teams across your organization.

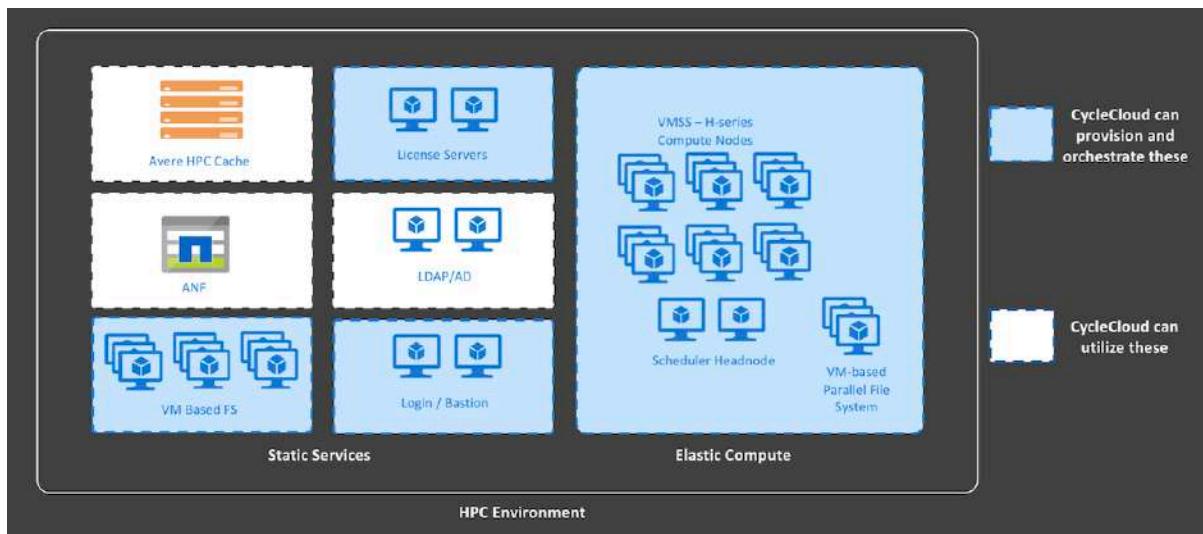
## What CycleCloud is Not?

There is no job scheduling functionality in CycleCloud. In other words, CycleCloud is not a scheduler, but rather a platform that enables users to deploy their own scheduler into Azure. CycleCloud comes with built-in support for a number of commonly used schedulers

(PBSPro, Slurm, IBM LSF, Grid Engine, and HT Condor), but CycleCloud users frequently implement their own scheduler on top of the provided autoscaling API.

CycleCloud does not dictate cluster topology; the installation comes with templates that are designed to get HPC systems up and running in Azure quickly, but HPC operators can customize these templates to tailor the infrastructure to meet their requirements. The Azure HPC community provides opinionated templates that are optimized for different types of workloads and industries.

## What a CycleCloud Deployed Environment Looks Like



An entire CycleCloud HPC system can be deployed on Azure infrastructure. CycleCloud itself is installed as an application server on a VM in Azure that requires outbound access to Azure Resource Provider APIs. CycleCloud then starts and manages VMs that form the HPC systems — these typically consist of the HPC scheduler head node(s) and compute nodes, but may also include VM based Network Attached Storage such as an NFS server or BeeGFS cluster, login nodes, bastion hosts, and other components needed to support an HPC infrastructure. The makeup of the HPC system is defined entirely through CycleCloud templates. Additionally, CycleCloud HPC environments can utilize other PaaS services such as Azure NetApp Files, Azure HPC Cache, and Azure Active Directory Domain Service.

## DEDICATED HOSTS

Azure Dedicated Host is a service that provides physical servers - able to host one or more virtual machines - dedicated to one Azure subscription. Dedicated hosts are the same physical servers used in our data centers, provided as a resource. You can provision dedicated hosts within a region, availability zone, and fault domain. Then, you can place VMs directly into your provisioned hosts, in whatever configuration best meets your needs.

### Benefits

Reserving the entire host provides the following benefits:

- Hardware isolation at the physical server level. No other VMs will be placed on your hosts. Dedicated hosts are deployed in the same data centers and share the same network and underlying storage infrastructure as other, non-isolated hosts.
- Control over maintenance events initiated by the Azure platform. While the majority of maintenance events have little to no impact on your virtual machines, there are some sensitive workloads where each second of pause can have an impact. With dedicated hosts, you can opt-in to a maintenance window to reduce the impact to your service.
- With the Azure hybrid benefit, you can bring your own licenses for Windows and SQL to Azure. Using the hybrid benefits provides you with additional benefits. For more information, see [Azure Hybrid Benefit](#).

### Groups, hosts, and VMs



A **host group** is a resource that represents a collection of dedicated hosts. You create a host group in a region and an availability zone, and add hosts to it.

A **host** is a resource, mapped to a physical server in an Azure data center. The physical server is allocated when the host is created. A host is created within a host group. A host has a SKU describing which VM sizes can be created. Each host can host multiple VMs, of different sizes, as long as they are from the same size series.

## **High Availability considerations**

For high availability, you should deploy multiple VMs, spread across multiple hosts (minimum of 2). With Azure Dedicated Hosts, you have several options to provision your infrastructure to shape your fault isolation boundaries.

### **Use Availability Zones for fault isolation**

Availability zones are unique physical locations within an Azure region. Each zone is made up of one or more datacenters equipped with independent power, cooling, and networking. A host group is created in a single availability zone. Once created, all hosts will be placed within that zone. To achieve high availability across zones, you need to create multiple host groups (one per zone) and spread your hosts accordingly.

If you assign a host group to an availability zone, all VMs created on that host must be created in the same zone.

### **Use Fault Domains for fault isolation**

A host can be created in a specific fault domain. Just like VM in a scale set or availability set, hosts in different fault domains will be placed on different physical racks in the data center. When you create a host group, you are required to specify the fault domain count. When creating hosts within the host group, you assign fault domain for each host. The VMs do not require any fault domain assignment.

Fault domains are not the same as colocation. Having the same fault domain for two hosts does not mean they are in proximity with each other.

Fault domains are scoped to the host group. You should not make any assumption on anti-affinity between two host groups (unless they are in different availability zones).

VMs deployed to hosts with different fault domains, will have their underlying managed disks services on multiple storage stamps, to increase the fault isolation protection.

## **Using Availability Zones and Fault Domains**

You can use both capabilities together to achieve even more fault isolation. In this case, you will specify the availability zone and fault domain count in for each host group, assign a fault domain to each of your hosts in the group, and assign an availability zone to each of your VMs

The Resource Manager sample template uses zones and fault domains to spread hosts for maximum resiliency in a region.

## **Manual vs. automatic placement**

When creating a VM in Azure, you can select which dedicated host to use. You can also use the option to automatically place your VMs on existing hosts, within a host group.

When creating a new host group, make sure the setting for automatic VM placement is selected. When creating your VM, select the host group and let Azure pick the best host for your VM.

Host groups that are enabled for automatic placement do not require all the VMs to be automatically placed. You will still be able to explicitly pick a host, even when automatic placement is selected for the host group.

## **Limitations**

Known issues and limitations when using automatic VM placement:

- You will not be able to redeploy your VM.
- You will not be able to use DCv2, Lsv2, NVasv4, NVsv3, Msv2, or M-series VMs with dedicated hosts

## **Virtual machine scale set support**

Virtual machine scale sets let you treat a group of virtual machines as a single resource, and apply availability, management, scaling and orchestration policies as a group. Your existing dedicated hosts can also be used for virtual machine scale sets.

When creating a virtual machine scale set you can specify an existing host group to have all of the VM instances created on dedicated hosts.

The following requirements apply when creating a virtual machine scale set in a dedicated host group:

- Automatic VM placement needs to be enabled.
- The availability setting of your host group should match your scale set.
  - A regional host group (created without specifying an availability zone) should be used for regional scale sets.

- The host group and the scale set must be using the same availability zone.
- The fault domain count for the host group level should match the fault domain count for your scale set. The Azure portal lets you specify *max spreading* for your scale set, which sets the fault domain count of 1.
- Dedicated hosts should be created first, with sufficient capacity, and the same settings for scale set zones and fault domains.
- The supported VM sizes for your dedicated hosts should match the one used for your scale set.

Not all scale-set orchestration and optimizations settings are supported by dedicated hosts. Apply the following settings to your scale set:

- Overprovisioning is not recommended, and it is disabled by default. You can enable overprovisioning, but the scale set allocation will fail if the host group does not have capacity for all of the VMs, including the overprovisioned instances.
- Use the ScaleSetVM orchestration mode
- Do not use proximity placement groups for co-location

## Maintenance control

The infrastructure supporting your virtual machines may occasionally be updated to improve reliability, performance, security, and to launch new features. The Azure platform tries to minimize the impact of platform maintenance whenever possible, but customers with *maintenance sensitive* workloads can't tolerate even few seconds that the VM needs to be frozen or disconnected for maintenance.

**Maintenance Control** provides customers with an option to skip regular platform updates scheduled on their dedicated hosts, then apply it at the time of their choice within a 35-day rolling window. Within the maintenance window, you can apply maintenance directly at the host level, in any order. Once the maintenance window is over, Microsoft will move forward and apply the pending maintenance to the hosts in an order which may not follow the user defined fault domains.

For more information, see [Managing platform updates with Maintenance Control](#).

## Capacity considerations

Once a dedicated host is provisioned, Azure assigns it to physical server. This guarantees the availability of the capacity when you need to provision your VM. Azure uses the entire capacity in the region (or zone) to pick a physical server for your host. It also means that

customers can expect to be able to grow their dedicated host footprint without the concern of running out of space in the cluster.

## Quotas

There are two types of quota that are consumed when you deploy a dedicated host.

1. Dedicated host vCPU quota. The default quota limit is 3000 vCPUs, per region.
2. VM size family quota. For example, a **Pay-as-you-go** subscription may only have a quota of 10 vCPUs available for the Dsv3 size series, in the East US region. To deploy a Dsv3 dedicated host, you would need to request a quota increase to at least 64 vCPUs before you can deploy the dedicated host.

To request a quota increase, create a support request in the Azure portal.

Provisioning a dedicated host will consume both dedicated host vCPU and the VM family vCPU quota, but it will not consume the regional vCPU. VMs placed on a dedicated host will not count against VM family vCPU quota. Should a VM be moved off a dedicated host into a multi-tenant environment, the VM will consume VM family vCPU quota.

Provider	Location	Usage
Microsoft.Compute	East US	0 % 0 of 10

For more information, see [Virtual machine vCPU quotas](#).

Free trial and MSDN subscriptions do not have quota for Azure Dedicated Hosts.

## Pricing

Users are charged per dedicated host, regardless how many VMs are deployed. In your monthly statement you will see a new billable resource type of hosts. The VMs on a dedicated host will still be shown in your statement, but will carry a price of 0.

The host price is set based on VM family, type (hardware size), and region. A host price is relative to the largest VM size supported on the host.

Software licensing, storage and network usage are billed separately from the host and VMs. There is no change to those billable items.

For more information, see [Azure Dedicated Host pricing](#).

You can also save on costs with a Reserved Instance of Azure Dedicated Hosts.

## Sizes and hardware generations

A SKU is defined for a host and it represents the VM size series and type. You can mix multiple VMs of different sizes within a single host as long as they are of the same size series.

The *type* is the hardware generation. Different hardware types for the same VM series will be from different CPU vendors and have different CPU generations and number of cores.

The sizes and hardware types vary by region. Refer to the host pricing page to learn more.

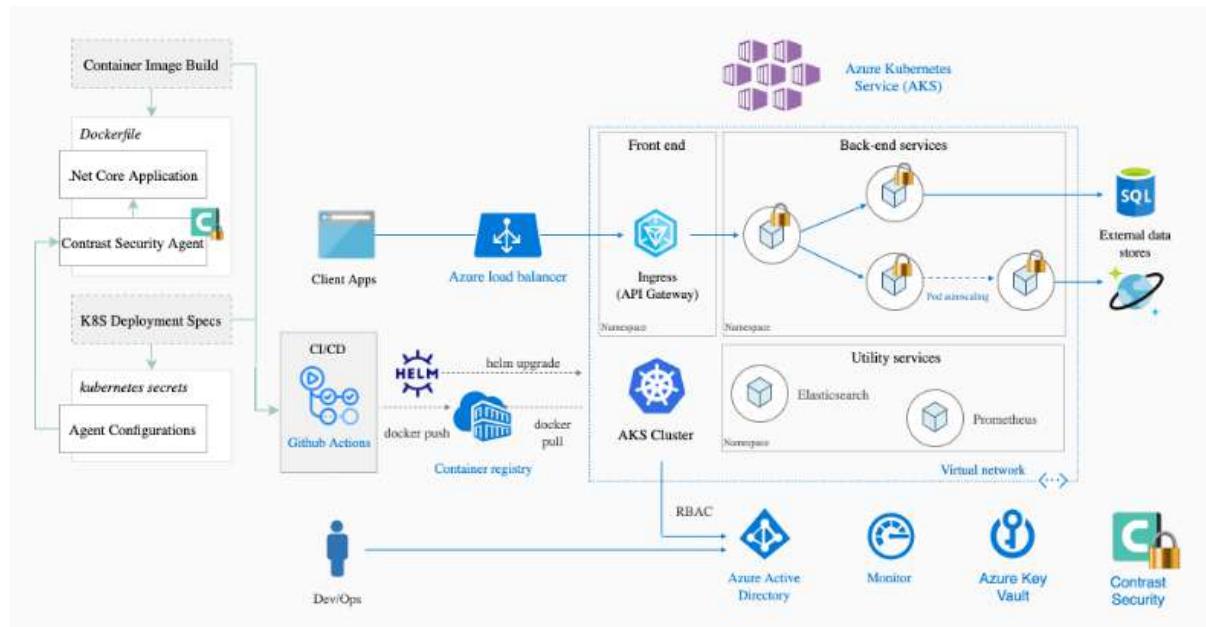
## Host life cycle

Azure monitors and manages the health status of your hosts. The following states will be returned when you query your host:

Health State	Description
Host Available	There are no known issues with your host.
Host Under Investigation	We're having some issues with the host which we're looking into. This is a transitional state required for Azure to try and identify the scope and root cause for the issue identified. Virtual machines running on the host may be
Host Pending Deallocate	Azure can't restore the host back to a healthy state and ask you to redeploy your virtual machines out of this host. If <code>autoReplaceOnFailure</code> is enabled, your virtual machines are <i>service healed</i> to healthy hardware. Otherwise, your virtual machine may be running on a host that is about to fail.
Host deallocated	All virtual machines have been removed from the host. You are no longer being charged for this host since the hardware was taken out of rotation.

# AZURE KUBERNETES SERVICES

Azure Kubernetes Service allows you to quickly deploy a production ready Kubernetes cluster in Azure. Azure Kubernetes Service (AKS) simplifies deploying a managed Kubernetes cluster in Azure by offloading the operational overhead to Azure. As a hosted Kubernetes service, Azure handles critical tasks, like health monitoring and maintenance. Since Kubernetes masters are managed by Azure, you only manage and maintain the agent nodes. Thus, AKS is free; you only pay for the agent nodes within your clusters, not for the masters.



You can create an AKS cluster using:

- The Azure CLI
- The Azure portal
- Azure PowerShell
- Using template-driven deployment options, like Azure Resource Manager templates and Terraform

When you deploy an AKS cluster, the Kubernetes master and all nodes are deployed and configured for you. Advanced networking, Azure Active Directory (Azure AD) integration, monitoring, and other features can be configured during the deployment process. AKS also supports Windows Server containers.

## Access, security, and monitoring

For improved security and management, AKS lets you integrate with Azure AD to:

- Use Kubernetes role-based access control (Kubernetes RBAC).

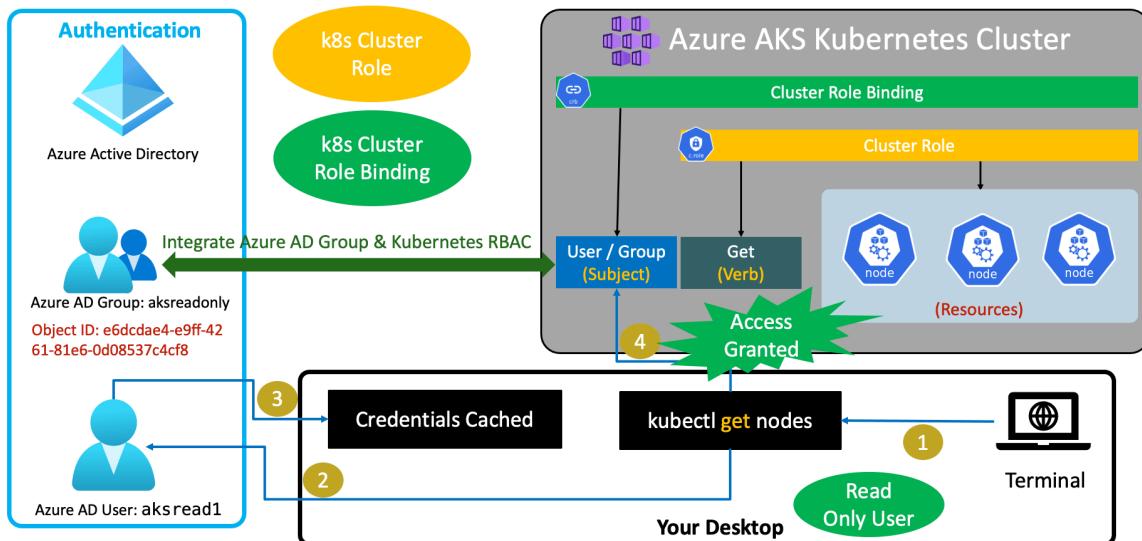
- Monitor the health of your cluster and resources.

## Identity and security management

### Kubernetes RBAC

To limit access to cluster resources, AKS supports Kubernetes RBAC. Kubernetes RBAC controls access and permissions to Kubernetes resources and namespaces.

## Azure Active Directory & Kubernetes RBAC



## Azure AD

You can configure an AKS cluster to integrate with Azure AD. With Azure AD integration, you can set up Kubernetes access based on existing identity and group membership. Your existing Azure AD users and groups can be provided with an integrated sign-on experience and access to AKS resources.

To secure your AKS clusters, see [Integrate Azure Active Directory with AKS](#).

## Integrated logging and monitoring

Azure Monitor for Container Health collects memory and processor performance metrics from containers, nodes, and controllers within your AKS cluster and deployed applications. You can review both container logs and the Kubernetes master logs, which are:

- Stored in an Azure Log Analytics workspace.
- Available through the Azure portal, Azure CLI, or a REST endpoint.

For more information, see [Monitor Azure Kubernetes Service container health](#).

## Clusters and nodes

AKS nodes run on Azure virtual machines (VMs). With AKS nodes, you can connect storage to nodes and pods, upgrade cluster components, and use GPUs. AKS supports Kubernetes clusters that run multiple node pools to support mixed operating systems and Windows Server containers.

## Cluster node and pod scaling

As demand for resources change, the number of cluster nodes or pods that run your services automatically scales up or down. You can adjust both the horizontal pod autoscaler or the cluster autoscaler to adjust to demands and only run necessary resources.

## Cluster node upgrades

AKS offers multiple Kubernetes versions. As new versions become available in AKS, you can upgrade your cluster using the Azure portal or Azure CLI. During the upgrade process, nodes are carefully cordoned and drained to minimize disruption to running applications.

## GPU-enabled nodes

AKS supports the creation of GPU-enabled node pools. Azure currently provides single or multiple GPU-enabled VMs. GPU-enabled VMs are designed for compute-intensive, graphics-intensive, and visualization workloads.

## Confidential computing nodes (public preview)

AKS supports the creation of Intel SGX-based, confidential computing node pools (DCSv2 VMs). Confidential computing nodes allow containers to run in a hardware-based, trusted execution environment (enclaves). Isolation between containers, combined with code integrity through attestation, can help with your defense-in-depth container security strategy.

Confidential computing nodes support both confidential containers (existing Docker apps) and enclave-aware containers.

## Storage volume support

To support application workloads, you can mount static or dynamic storage volumes for persistent data. Depending on the number of connected pods expected to share the storage volumes, you can use storage backed by either:

- Azure Disks for single pod access, or
- Azure Files for multiple, concurrent pod access.

## Virtual networks and ingress

An AKS cluster can be deployed into an existing virtual network. In this configuration, every pod in the cluster is assigned an IP address in the virtual network, and can directly communicate with:

- Other pods in the cluster
- Other nodes in the virtual network.

Pods can also connect to other services in a peered virtual network and to on-premises networks over ExpressRoute or site-to-site (S2S) VPN connections.

## Ingress with HTTP application routing

The HTTP application routing add-on helps you easily access applications deployed to your AKS cluster. When enabled, the HTTP application routing solution configures an ingress controller in your AKS cluster.

As applications are deployed, publicly accessible DNS names are autoconfigured. The HTTP application routing sets up a DNS zone and integrates it with the AKS cluster. You can then deploy Kubernetes ingress resources as normal.

## Development tooling integration

Kubernetes has a rich ecosystem of development and management tools that work seamlessly with AKS. These tools include Helm and the Kubernetes extension for Visual Studio Code.

Azure provides several tools that help streamline Kubernetes, such as DevOps Starter.

## **DevOps Starter**

DevOps Starter provides a simple solution for bringing existing code and Git repositories into Azure. DevOps Starter automatically:

- Creates Azure resources (such as AKS);
- Configures a release pipeline in Azure DevOps Services that includes a build pipeline for CI;
- Sets up a release pipeline for CD; and,
- Generates an Azure Application Insights resource for monitoring.

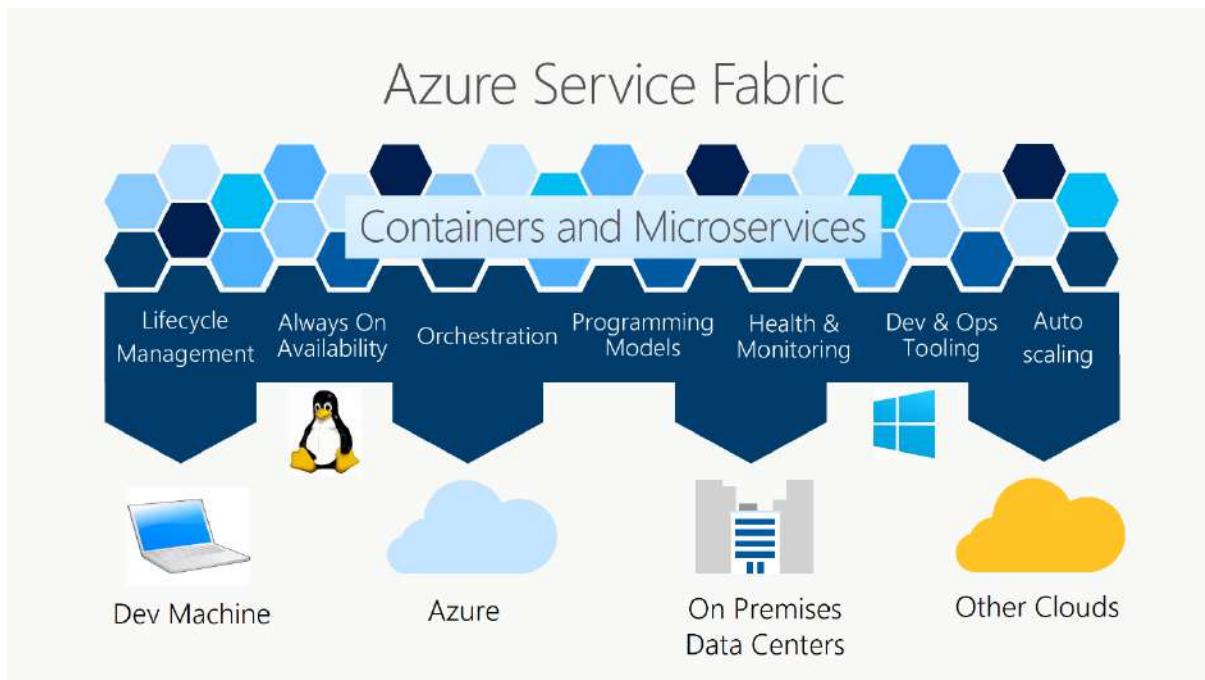
## **Docker image support and private container registry**

AKS supports the Docker image format. For private storage of your Docker images, you can integrate AKS with Azure Container Registry (ACR).

## **SERVICE FABRIC**

Azure Service Fabric is a distributed systems platform that makes it easy to package, deploy, and manage scalable and reliable microservices and containers. Service Fabric also addresses the significant challenges in developing and managing cloud native applications.

A key differentiator of Service Fabric is its strong focus on building stateful services. You can use the Service Fabric programming model or run containerized stateful services written in any language or code. You can create Service Fabric clusters anywhere, including Windows Server and Linux on premises and other public clouds, in addition to Azure.



Service Fabric powers many Microsoft services today, including Azure SQL Database, Azure Cosmos DB, Cortana, Microsoft Power BI, Microsoft Intune, Azure Event Hubs, Azure IoT Hub, Dynamics 365, Skype for Business, and many core Azure services.

## Container orchestration

Service Fabric is Microsoft's container orchestrator for deploying and managing microservices across a cluster of machines, benefiting from the lessons learned running Microsoft services at massive scale. Service Fabric can deploy applications in seconds, at high density with hundreds or thousands of applications or containers per machine. With Service Fabric, you can mix both services in processes and services in containers in the same application.

## Stateless and stateful microservices

Service Fabric provides a sophisticated, lightweight runtime that supports stateless and stateful microservices. A key differentiator of Service Fabric is its robust support for building stateful services, either with Service Fabric built-in programming models or containerized stateful services.

## **Application lifecycle management**

Service Fabric provides support for the full application lifecycle and CI/CD of cloud applications including containers: development through deployment, daily monitoring, management, and maintenance, to eventual decommissioning. Service Fabric is integrated with CI/CD tools such as Azure Pipelines, Jenkins, and Octopus Deploy and can be used with any other popular CI/CD tool.

## **Any OS, any cloud**

You can create clusters for Service Fabric in many environments, including Azure or on premises, on Windows Server or Linux. You can even create clusters on other public clouds. The development environment in the Service Fabric SDK is identical to the production environment, with no emulators involved. In other words, what runs on your local development cluster is what deploys to your clusters in other environments.

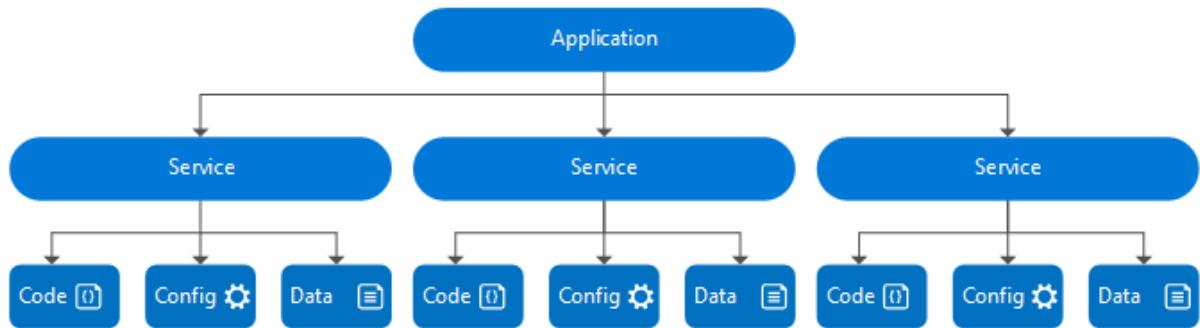
For Windows development, the Service Fabric .NET SDK is integrated with Visual Studio and PowerShell. For Linux development, the Service Fabric Java SDK is integrated with Eclipse, and Yeoman is used to generate templates for Java, .NET Core, and container applications.

## **Compliance**

Azure Service Fabric Resource Provider is available in all Azure regions and is compliant with all Azure compliance certifications, including: SOC, ISO, PCI DSS, HIPAA, and GDPR. For a complete list, see [Microsoft Compliance Offerings](#).

## **Model an application in Service Fabric**

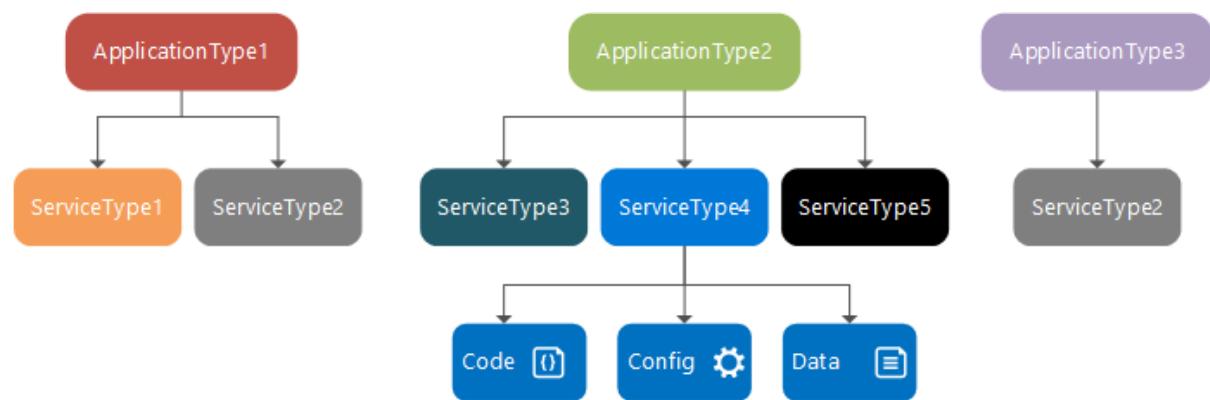
An application is a collection of constituent services that perform a certain function or functions. A service performs a complete and standalone function and can start and run independently of other services. A service is composed of code, configuration, and data. For each service, code consists of the executable binaries, configuration consists of service settings that can be loaded at run time, and data consists of arbitrary static data to be consumed by the service. Each component in this hierarchical application model can be versioned and upgraded independently.



An application type is a categorization of an application and consists of a bundle of service types. A service type is a categorization of a service. The categorization can have different settings and configurations, but the core functionality remains the same. The instances of a service are the different service configuration variations of the same service type.

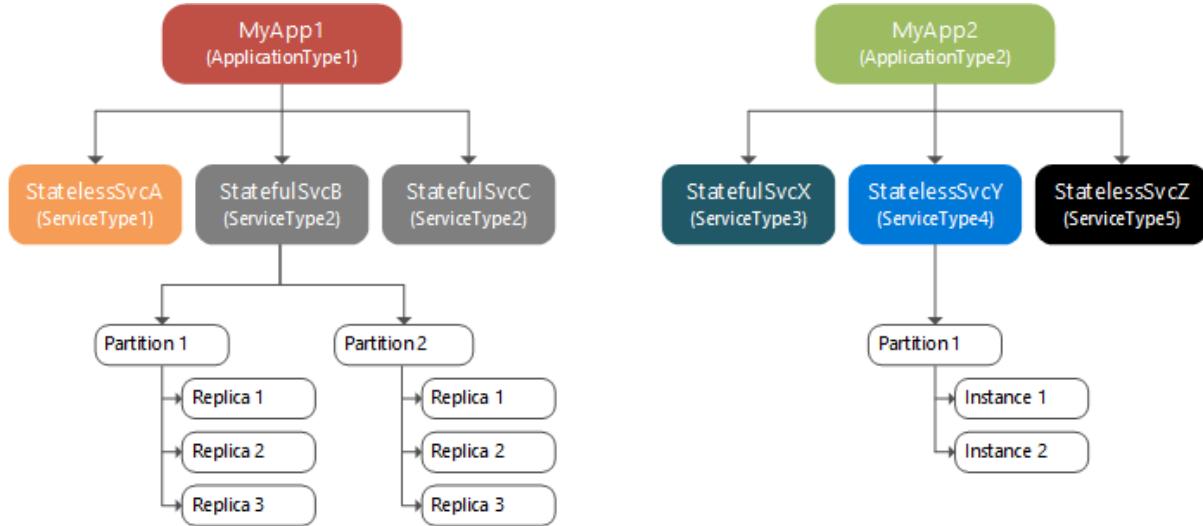
Classes (or "types") of applications and services are described through XML files (application manifests and service manifests). The manifests describe applications and services and are the templates against which applications can be instantiated from the cluster's image store. Manifests are covered in detail in Application and service manifests. The schema definition for the `ServiceManifest.xml` and `ApplicationManifest.xml` file is installed with the Service Fabric SDK and tools to `C:\Program Files\Microsoft SDKs\Service Fabric\schemas\ServiceFabricServiceModel.xsd`. The XML schema is documented in `ServiceFabricServiceModel.xsd` schema documentation.

The code for different application instances runs as separate processes even when hosted by the same Service Fabric node. Furthermore, the lifecycle of each application instance can be managed (for example, upgraded) independently. The following diagram shows how application types are composed of service types, which in turn are composed of code, configuration, and data packages. To simplify the diagram, only the code/config/data packages for ServiceType4 are shown, though each service type would include some or all those package types.



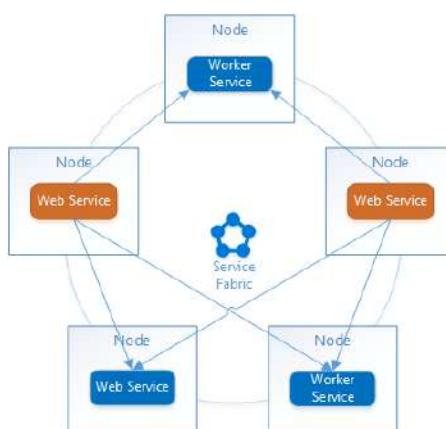
There can be one or more instances of a service type active in the cluster. For example, stateful service instances, or replicas, achieve high reliability by replicating state between replicas located on different nodes in the cluster. Replication essentially provides redundancy for the service to be available even if one node in a cluster fails. A partitioned service further divides its state (and access patterns to that state) across nodes in the cluster.

The following diagram shows the relationship between applications and service instances, partitions, and replicas.



## Service Fabric terminology overview

Azure Service Fabric is a distributed systems platform that makes it easy to package, deploy, and manage scalable and reliable microservices. Service Fabric is a container and process orchestrator that allows you to host your clusters anywhere: on Azure, in an on-premises datacenter, or on any cloud provider. You can use any framework to write your services and choose where to run the application from multiple environment choices. This article details the terminology used by Service Fabric to understand the terms used in the documentation.



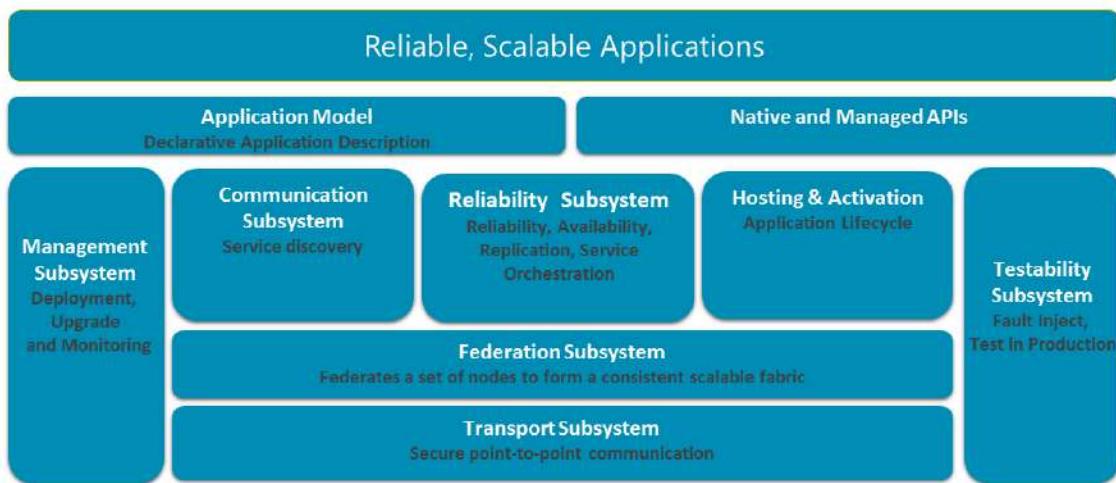
## Infrastructure concepts

**Cluster:** A network-connected set of virtual or physical machines into which your microservices are deployed and managed. Clusters can scale to thousands of machines.

**Node:** A machine or VM that's part of a cluster is called a *node*. Each node is assigned a node name (string). Nodes have characteristics, such as placement properties. Each machine or VM has an auto-start Windows service, FabricHost.exe, that starts running upon boot and then starts two executables: Fabric.exe and FabricGateway.exe. These two executables make up the node. For testing scenarios, you can host multiple nodes on a single machine or VM by running multiple instances of Fabric.exe and FabricGateway.exe.

## Application and service concepts

**Service Fabric Native Application:** Service Fabric Native Applications are described by the Native Application Model (XML-based application and service manifests).



## Service Fabric Native Application concepts

**Application:** An application is a collection of constituent services that perform a certain function or functions. The lifecycle of each application instance can be managed independently.

**Service:** A service performs a complete and standalone function and can start and run independently of other services. A service is composed of code, configuration, and data. For

each service, code consists of the executable binaries, configuration consists of service settings that can be loaded at run time, and data consists of arbitrary static data to be consumed by the service.

**Application type:** The name/version assigned to a collection of service types. It is defined in an ApplicationManifest.xml file and embedded in an application package directory. The directory is then copied to the Service Fabric cluster's image store. You can then create a named application from this application type within the cluster..

**Application package:** A disk directory containing the application type's ApplicationManifest.xml file. References the service packages for each service type that makes up the application type. The files in the application package directory are copied to Service Fabric cluster's image store. For example, an application package for an email application type might contain references to a queue-service package, a frontend-service package, and a database-service package.

**Named application:** After you copy an application package to the image store, you create an instance of the application within the cluster. You create an instance when you specify the application package's application type, by using its name or version. Each application type instance is assigned a uniform resource identifier (URI) name that looks like: "fabric:/MyNamedApp". Within a cluster, you can create multiple named applications from a single application type. You can also create named applications from different application types. Each named application is managed and versioned independently.

**Service type:** The name/version assigned to a service's code packages, data packages, and configuration packages. The service type is defined in the ServiceManifest.xml file and embedded in a service package directory. The service package directory is then referenced by an application package's ApplicationManifest.xml file. Within the cluster, after creating a named application, you can create a named service from one of the application type's service types. The service type's ServiceManifest.xml file describes the service.

There are two types of services:

- **Stateless:** Use a stateless service when the service's persistent state is stored in an external storage service, such as Azure Storage, Azure SQL Database, or Azure Cosmos DB. Use a stateless service when the service has no persistent storage. For example, for a calculator service where values are passed to the service, a computation is performed that uses these values, and then a result is returned.
- **Stateful:** Use a stateful service when you want Service Fabric to manage your service's state via its Reliable Collections or Reliable Actors programming models. When you create a named service, specify how many partitions you want to spread your state over for scalability. Also specify how many times to replicate your state across nodes, for reliability. Each named service has a single primary replica and multiple secondary replicas. You modify your named service's state when you write to

the primary replica. Service Fabric then replicates this state to all the secondary replicas to keep your state in sync. Service Fabric automatically detects when a primary replica fails and promotes an existing secondary replica to a primary replica. Service Fabric then creates a new secondary replica.

**Replicas or instances** refer to code (and state for stateful services) of a service that's deployed and running. See [Replicas and instances](#).

**Reconfiguration** refers to the process of any change in the replica set of a service. See [Reconfiguration](#).

**Service package:** A disk directory containing the service type's ServiceManifest.xml file. This file references the code, static data, and configuration packages for the service type. The files in the service package directory are referenced by the application type's ApplicationManifest.xml file. For example, a service package might refer to the code, static data, and configuration packages that make up a database service.

**Named service:** After you create a named application, you can create an instance of one of its service types within the cluster. You specify the service type by using its name/version. Each service type instance is assigned a URI name scoped under its named application's URI. For example, if you create a "MyDatabase" named service within a "MyNamedApp" named application, the URI looks like: "fabric:/MyNamedApp/MyDatabase". Within a named application, you can create several named services. Each named service can have its own partition scheme and instance or replica counts.

**Code package:** A disk directory containing the service type's executable files, typically EXE/DLL files. The files in the code package directory are referenced by the service type's ServiceManifest.xml file. When you create a named service, the code package is copied to the node or nodes selected to run the named service. Then the code starts to run. There are two types of code package executables:

- **Guest executables:** Executables that run as-is on the host operating system (Windows or Linux). These executables don't link to or reference any Service Fabric runtime files and therefore don't use any Service Fabric programming models. These executables are unable to use some Service Fabric features, such as the naming service for endpoint discovery. Guest executables can't report load metrics that are specific to each service instance.

- **Service host executables:** Executables that use Service Fabric programming models by linking to Service Fabric runtime files, enabling Service Fabric features. For example, a named service instance can register endpoints with Service Fabric's Naming Service and can also report load metrics.

**Data package:** A disk directory that contains the service type's static, read-only data files, typically photo, sound, and video files. The files in the data package directory are referenced by the service type's ServiceManifest.xml file. When you create a named service, the data package is copied to the node or nodes selected to run the named service. The code starts running and can now access the data files.

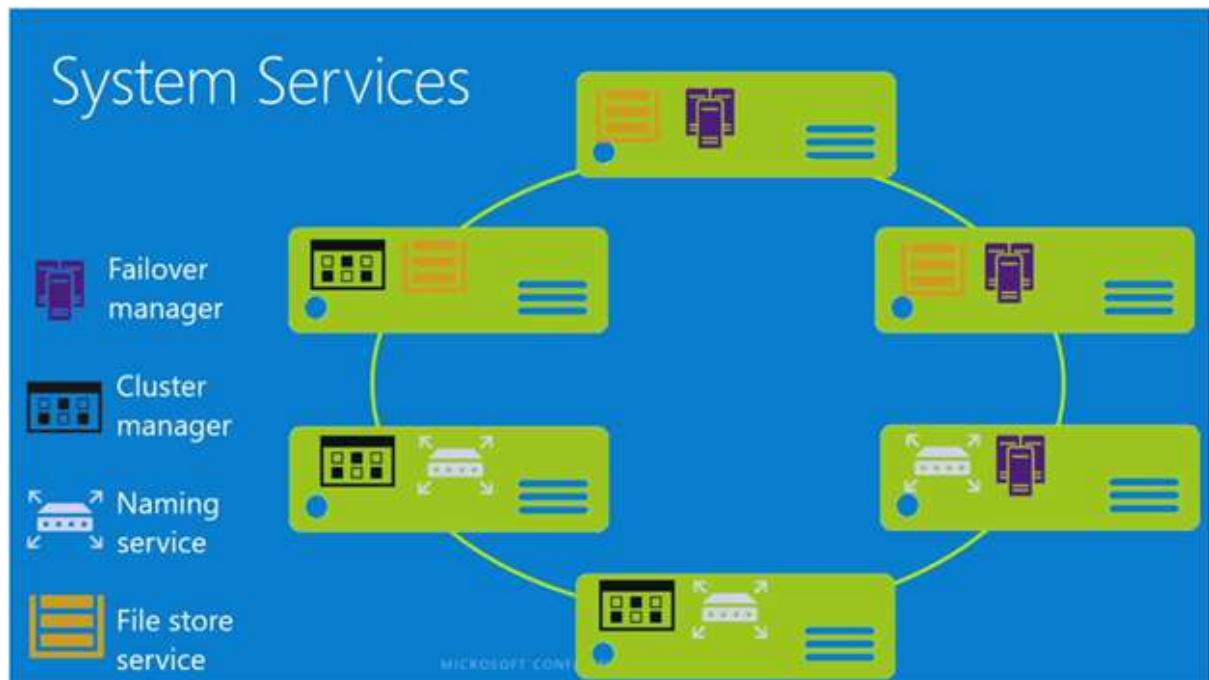
**Configuration package:** A disk directory that contains the service type's static, read-only configuration files, typically text files. The files in the configuration package directory are referenced by the service type's ServiceManifest.xml file. When you create a named service, the files in the configuration package are copied to one or more nodes selected to run the named service. Then the code starts to run and can now access the configuration files.

**Containers:** By default, Service Fabric deploys and activates services as processes. Service Fabric can also deploy services in container images. Containers are a virtualization technology that abstracts the underlying operating system from applications. An application and its runtime, dependencies, and system libraries run inside a container. The container has full, private access to the container's own isolated view of the operating system constructs. Service Fabric supports Windows Server containers and Docker containers on Linux. For more information, read Service Fabric and containers.

**Partition scheme:** When you create a named service, you specify a partition scheme. Services with substantial amounts of state split the data across partitions, which spreads the state across the cluster's nodes. By splitting the data across partitions, your named service's state can scale. Within a partition, stateless named services have instances, whereas stateful named services have replicas. Usually, stateless named services have only one partition, because they have no internal state. The partition instances provide for availability. If one instance fails, other instances continue to operate normally, and then Service Fabric creates a new instance. Stateful named services maintain their state within replicas and each partition has its own replica set so the state is kept in sync. Should a replica fail, Service Fabric builds a new replica from the existing replicas.

## System services

There are system services that are created in every cluster that provide the platform capabilities of Service Fabric.



**Naming Service:** Each Service Fabric cluster has a Naming Service, which resolves service names to a location in the cluster. You manage the service names and properties, like an internet Domain Name System (DNS) for the cluster. Clients securely communicate with any node in the cluster by using the Naming Service to resolve a service name and its location. Applications move within the cluster. For example, this can be due to failures, resource balancing, or the resizing of the cluster. You can develop services and clients that resolve the current network location. Clients obtain the actual machine IP address and port where it's currently running.

**Image Store service:** Each Service Fabric cluster has an Image Store service where deployed, versioned application packages are kept. Copy an application package to the Image Store and then register the application type contained within that application package. After the application type is provisioned, you create a named application from it. You can unregister an application type from the Image Store service after all its named applications have been deleted.

**Failover Manager service:** Each Service Fabric cluster has a Failover Manager service that is responsible for the following actions:

- Performs functions related to high availability and consistency of services.
- Orchestrates application and cluster upgrades.
- Interacts with other system components.

**Repair Manager service:** This is an optional system service that allows repair actions to be performed on a cluster in a way that is safe, automatable, and transparent. Repair manager is used in:

- Performing Azure maintenance repairs on Silver and Gold durability Azure Service Fabric clusters.
- Carrying out repair actions for Patch Orchestration Application

## Deployment and application models

To deploy your services, you need to describe how they should run. Service Fabric supports three different deployment models:

### Native model

The native application model provides your applications with full low-level access to Service Fabric. Applications and services are defined as registered types in XML manifest files.

The native model supports the Reliable Services and Reliable Actors frameworks, which provides access to the Service Fabric runtime APIs and cluster management APIs in C# and Java. The native model also supports arbitrary containers and executables.

**Reliable Services:** An API to build stateless and stateful services. Stateful services store their state in Reliable Collections, such as a dictionary or a queue. You can also plug in various communication stacks, such as Web API and Windows Communication Foundation (WCF).

**Reliable Actors:** An API to build stateless and stateful objects through the virtual Actor programming model. This model is useful when you have lots of independent units of computation or state. This model uses a turn-based threading model, so it's best to avoid code that calls out to other actors or services because an individual actor can't process other incoming requests until all its outbound requests are finished.

You can also run your existing applications on Service Fabric:

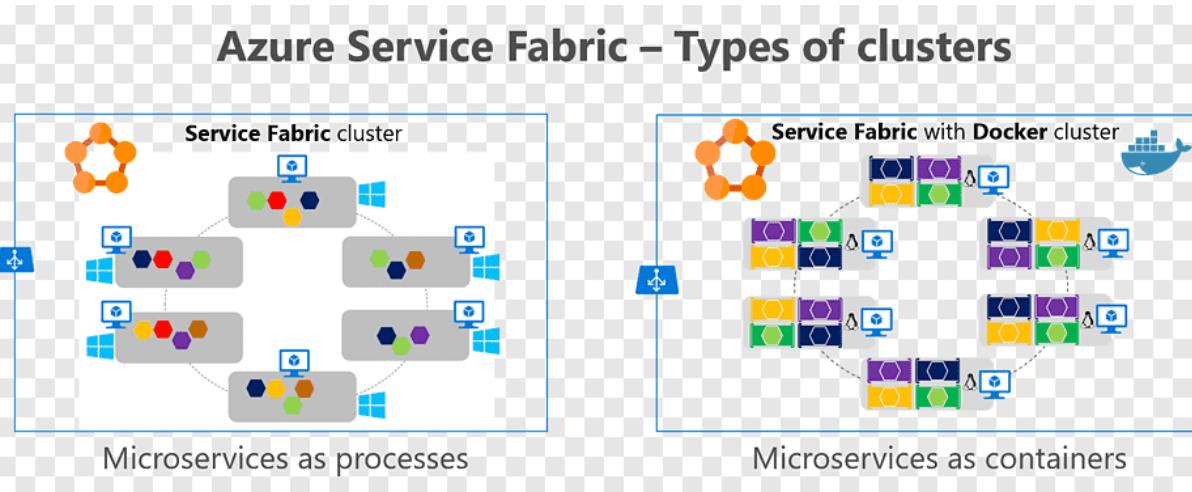
**Containers:** Service Fabric supports the deployment of Docker containers on Linux and Windows Server containers on Windows Server 2016, along with support for Hyper-V isolation mode. In the Service Fabric application model, a container represents an application host in which multiple service replicas are placed. Service Fabric can run any containers, and the scenario is similar to the guest executable scenario, where you package an existing application inside a container. In addition, you can run Service Fabric services inside containers as well.

**Guest executables:** You can run any type of code, such as Node.js, Python, Java, or C++ in Azure Service Fabric as a service. Service Fabric refers to these types of services as guest executables, which are treated as stateless services. The advantages to running a guest

executable in a Service Fabric cluster include high availability, health monitoring, application lifecycle management, high density, and discoverability.

## Docker Compose

Docker Compose is part of the Docker project. Service Fabric provides limited support for deploying applications using the Docker Compose model.



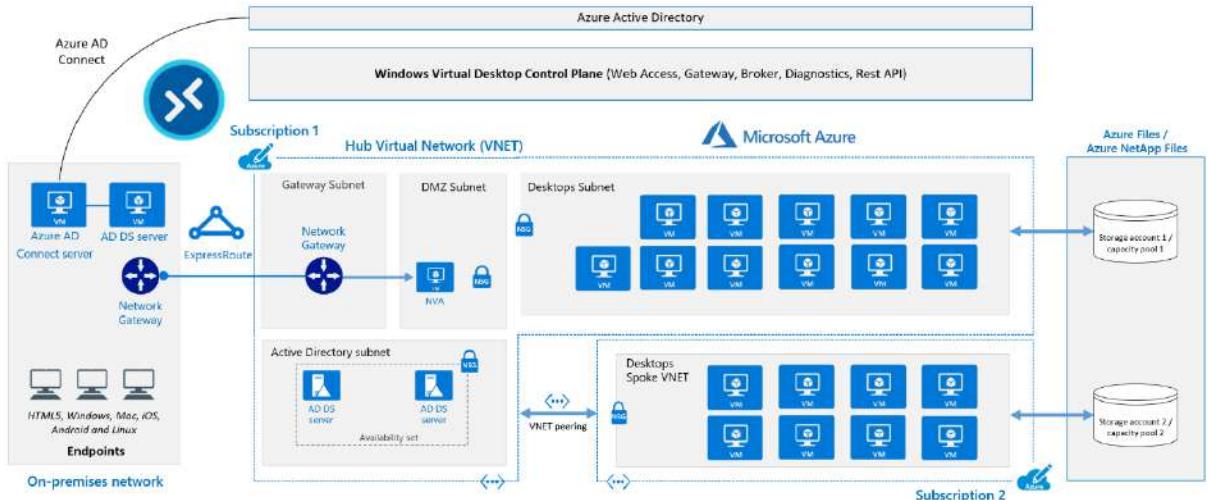
## Environments

Service Fabric is an open-source platform technology that several different services and products are based on. Microsoft provides the following options:

- **Azure Service Fabric:** The Azure hosted Service Fabric cluster offering. It provides integration between Service Fabric and the Azure infrastructure, along with upgrade and configuration management of Service Fabric clusters.
- **Service Fabric standalone:** A set of installation and configuration tools to deploy Service Fabric clusters anywhere (on-premises or on any cloud provider). Not managed by Azure.
- **Service Fabric development cluster:** Provides a local development experience on Windows, Linux, or Mac for development of Service Fabric applications.

# AZURE VIRTUAL DESKTOP

Azure Virtual Desktop is a desktop and app virtualization service that runs on the cloud.



Azure Virtual Desktop can:

- Set up a multi-session Windows 10 deployment that delivers a full Windows 10 with scalability
- Virtualize Microsoft 365 Apps for enterprise and optimize it to run in multi-user virtual scenarios
- Provide Windows 7 virtual desktops with free Extended Security Updates
- Bring your existing Remote Desktop Services (RDS) and Windows Server desktops and apps to any computer
- Virtualize both desktops and apps
- Manage Windows 10, Windows Server, and Windows 7 desktops and apps with a unified management experience

## Key capabilities

With Azure Virtual Desktop, you can set up a scalable and flexible environment:

- Create a full desktop virtualization environment in your Azure subscription without running any gateway servers.
- Publish as many host pools as you need to accommodate your diverse workloads.
- Bring your own image for production workloads or test from the Azure Gallery.

- Reduce costs with pooled, multi-session resources. With the new Windows 10 Enterprise multi-session capability, exclusive to Azure Virtual Desktop and Remote Desktop Session Host (RDSH) role on Windows Server, you can greatly reduce the number of virtual machines and operating system (OS) overhead while still providing the same resources to your users.
- Provide individual ownership through personal (persistent) desktops.

You can deploy and manage virtual desktops:

- Use the Azure portal, Azure Virtual Desktop PowerShell and REST interfaces to configure the host pools, create app groups, assign users, and publish resources.
- Publish full desktop or individual remote apps from a single host pool, create individual app groups for different sets of users, or even assign users to multiple app groups to reduce the number of images.
- As you manage your environment, use built-in delegated access to assign roles and collect diagnostics to understand various configuration or user errors.
- Use the new Diagnostics service to troubleshoot errors.
- Only manage the image and virtual machines, not the infrastructure. You don't need to personally manage the Remote Desktop roles like you do with Remote Desktop Services, just the virtual machines in your Azure subscription.

You can also assign and connect users to your virtual desktops:

- Once assigned, users can launch any Azure Virtual Desktop client to connect to their published Windows desktops and applications. Connect from any device through either a native application on your device or the Azure Virtual Desktop HTML5 web client.
- Securely establish users through reverse connections to the service, so you never have to leave any inbound ports open.

## **Set up Azure Virtual Desktop manually**

You can set up your deployment manually by following these steps:

### **1. Create a host pool with the Azure portal**

Host pools are a collection of one or more identical virtual machines (VMs), also known as "session hosts," within Azure Virtual Desktop environments. Each host pool

can contain an app group that users can interact with as they would on a physical desktop. If you'd like to learn more about deployment architecture, check out Azure Virtual Desktop environment. If you're an app developer using remote app streaming for Azure Virtual Desktop, your customers or users can use your apps just like local apps on a physical device.

### **Begin the host pool setup process**

To start creating your new host pool:

1. Sign in to the Azure portal at <https://portal.azure.com>.
2. Enter **Azure Virtual Desktop** into the search bar, then find and select **Azure Virtual Desktop** under Services.
3. In the **Azure Virtual Desktop** overview page, select **Create a host pool**.
4. In the **Basics** tab, select the correct subscription under Project details.
5. Either select **Create new** to make a new resource group or select an existing resource group from the drop-down menu.
6. Enter a unique name for your host pool.
7. In the Location field, select the region where you want to create the host pool from the drop-down menu.

The Azure geography associated with the regions you selected is where the metadata for this host pool and its related objects will be stored. Make sure you choose the regions inside the geography you want the service metadata to be stored in.

Project details

Subscription *	<input type="text" value="Microsoft Azure"/>
Resource group *	<input type="text" value="Select a resource group"/> <a href="#">Create new</a>
Host pool name *	<input type="text"/>
Location *	<input type="text" value="(US) East US"/> <small>Metadata will be stored in East US</small>

8. Under Host pool type, select whether your host pool will be **Personal** or **Pooled**.
  - If you choose **Personal**, then select either **Automatic** or **Direct** in the Assignment Type field.

Host pool type *	Personal	⌄
Assignment type ⓘ	Automatic	⌃
	Automatic	
	Direct	

9. If you choose **Pooled**, enter the following information:

- For **Max session limit**, enter the maximum number of users you want load-balanced to a single session host.
- For **Load balancing algorithm**, choose either breadth-first or depth-first, based on your usage pattern. Learn more about what each of these options means at Host pool load-balancing methods.

Host pool type *	Pooled	⌄
Max session limit ⓘ	Max # of users	
Load balancing algorithm ⓘ	Breadth-first	⌃
	Breadth-first	
	Depth-first	

10. Select **Next: Virtual Machines >**.

11. If you've already created virtual machines and want to use them with the new host pool, select **No**, select **Next: Workspace >** and jump to the Workspace information section. If you want to create new virtual machines and register them to the new host pool, select **Yes**.

Now that you've created a host pool, let's move on to the next part of the setup process where we create the VM.

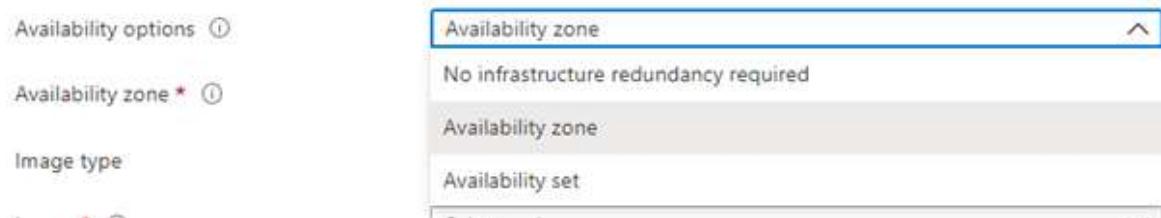
### Virtual machine details

Now that we're through the first part, you'll have to set up your VM.

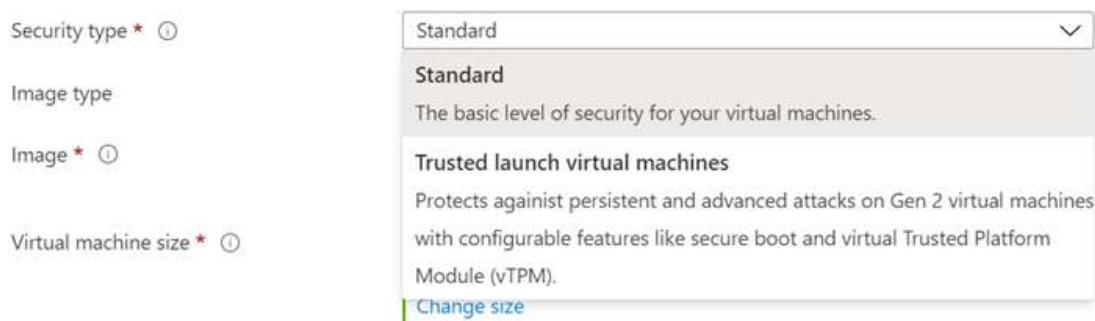
To set up your virtual machine within the Azure portal host pool setup process:

1. Under **Resource group**, choose the resource group where you want to create the virtual machines. This can be a different resource group than the one you used for the host pool.
2. After that, provide a **Name prefix** to name the virtual machines the setup process creates. The suffix will be - with numbers starting from 0.

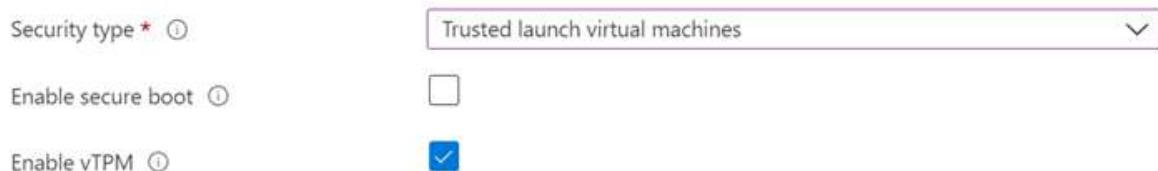
3. Choose the **Virtual machine location** where you want to create the virtual machines. They can be the same or different from the region you selected for the host pool. Keep in mind that VM prices vary by region, and the VM locations should be near their users when possible to maximize performance. Learn more at Data locations for Azure Virtual Desktop.
4. Next, choose the availability option that best suit your needs. To learn more about which option is right for you, see Availability options for virtual machines in Azure and our FAQ.



5. Next, choose the security type that you would like to use for your virtual machines. You can choose either **Standard** or **Trusted Launch virtual machines**. To learn more about Trusted Launch virtual machines, see Trusted Launch for Azure virtual machines.



If Trusted Launch virtual machines is selected, choose which Trusted Launch security features you would like to enable.



6. Next, choose the image that needs to be used to create the virtual machine. You can choose either **Gallery** or **Storage blob**.

- If you choose **Gallery**, select one of the recommended images from the drop-down menu:
  - Windows 10 Enterprise multi-session, Version 1909

- Windows 10 Enterprise multi-session, Version 1909 + Microsoft 365 Apps
- Windows Server 2019 Datacenter
- Windows 10 Enterprise multi-session, Version 2004
- Windows 10 Enterprise multi-session, Version 2004 + Microsoft 365 Apps

If you don't see the image you want, select **See all images**, which lets you select either another image in your gallery or an image provided by Microsoft and other publishers. Make sure that the image you choose is one of the supported OS images.

Select an image

Marketplace My Items

AI + Machine Learning

Analytics

Blockchain

Compute

Containers

Databases

Developer Tools

DevOps

Identity

Integration

Internet of Things

windows server datacenter

	<a href="#">Server Datacenter Core</a> Microsoft PreRTM image of Windows Server Semi-Annual Channel for validation purposes only
	<a href="#">Windows Server 2019 Datacenter (zh-cn)</a> Microsoft
	<a href="#">Windows Server 2012 R2 Datacenter (zh-cn)</a> Microsoft Enterprise-class solutions that are simple to deploy, cost-effective, application-focused, and user-centric.
	<a href="#">Windows Server 2012 Datacenter (zh-cn)</a> Microsoft Simple to deploy, cost-effective, application-focused, and user-centric.

You can also go to **My Items** and choose a custom image you've already uploaded.

Select an image

Marketplace My Items

My Images

Search

Shared Images

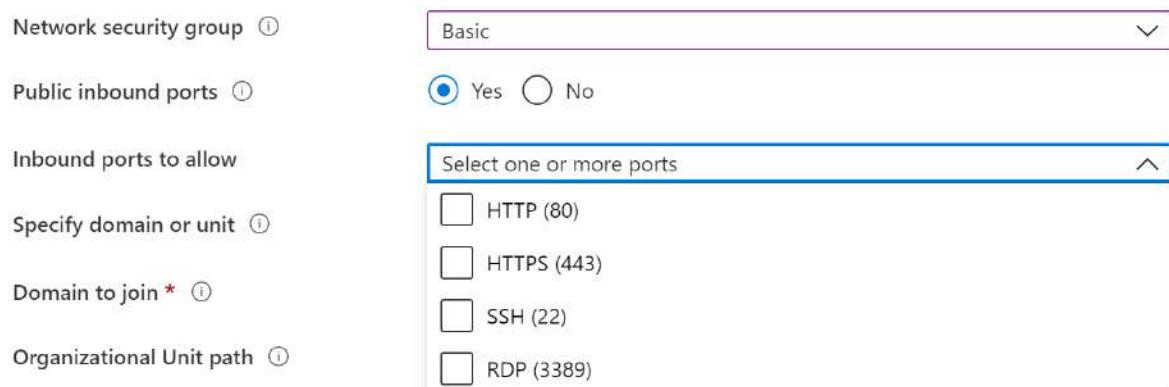
No results

- If you choose **Storage Blob**, you can use your own image build through Hyper-V or on an Azure VM. All you have to do is enter the location of the image in the storage blob as a URI.

The image's location is independent of the availability option, but the image's zone resiliency determines whether that image can be used with availability zone. If you select an availability zone while creating your image, make sure you're using an image from the gallery with zone resiliency enabled. To learn more about which zone resiliency option you should use, see the FAQ.

7. After that, choose the **Virtual machine size** you want to use. You can either keep the default size as-is or select **Change size** to change the size. If you select **Change size**, in the window that appears, choose the size of the virtual machine suitable for your workload. To learn more about virtual machine sizes and which size you should choose, see [Virtual machine sizing guidelines](#).
8. Under **Number of VMs**, provide the number of VMs you want to create for your host pool.
9. Choose what kind of OS disks you want your VMs to use: Standard SSD, Premium SSD, or Standard HDD.
10. Under Network and security, select the **Virtual network** and **Subnet** where you want to put the virtual machines you create. Make sure the virtual network can connect to the domain controller, since you'll need to join the virtual machines inside the virtual network to the domain. The DNS servers of the virtual network you selected should be configured to use the IP of the domain controller.
11. Select what kind of security group you want: **Basic**, **Advanced**, or **None**.

If you select **Basic**, you'll have to select whether you want any inbound port open. If you select **Yes**, choose from the list of standard ports to allow inbound connections to.



If you choose **Advanced**, select an existing network security group that you've already configured.

12. After that, select whether you want the virtual machines to be joined to **Active Directory** or **Azure Active Directory (Preview)**.

- For Active Directory, provide an account to join the domain and choose if you want to join a specific domain and organizational unit.
  - For the AD domain join UPN, enter the credentials for the Active Directory Domain admin of the virtual network you selected. The account you use can't have multifactor authentication (MFA) enabled. When joining to an Azure Active Directory Domain Services (Azure AD DS) domain, the account you use must be part of the Azure AD DC Administrators group and the account password must work in Azure AD DS.
  - To specify a domain, select **Yes**, then enter the name of the domain you want to join. If you want, you can also add a specific organizational unit you want the virtual machines to be in by entering the full path (Distinguished Name) and without quotation marks. If you don't want to specify a domain, select **No**. The VMs will automatically join the domain that matches the suffix of the **AD domain join UPN**.
- For Azure Active Directory, you can select **Enroll the VM with Intune** to automatically make the VM available for management after it's deployed.

13. Under **Virtual Machine Administrator account**, enter the credentials for the local admin account to be added while creating the VM. You can use this account for management purposes in both AD and Azure AD-joined VMs.

14. Under **Post update custom configuration**, you can enter the location of an Azure Resource Manager template to perform custom configurations on your session hosts after you create them. You'll need to enter the URLs for both the Azure Resource Manager template file and the Azure Resource Manager template parameter file.

15. Select **Next: Workspace >**.

With that, we're ready to start the next phase of setting up your host pool: registering your app group to a workspace.

## **Workspace information**

The host pool setup process creates a desktop application group by default. For the host pool to work as intended, you'll need to publish this app group to users or user groups, and you must register the app group to a workspace.

To register the desktop app group to a workspace:

1. Select **Yes**.

If you select **No**, you can register the app group later, but we recommend you get the workspace registration done as soon as you can so your host pool works properly.

2. Next, choose whether you want to create a new workspace or select from existing workspaces. Only workspaces created in the same location as the host pool will be allowed to register the app group to.
3. Optionally, you can select **Next: Tags >**.

Here you can add tags so you can group the objects with metadata to make things easier for your admins.

4. When you're done, select **Review + create**.
5. Review the information about your deployment to make sure everything looks correct. When you're done, select **Create**.

This starts the deployment process, which creates the following objects:

- Your new host pool.
- A desktop app group.
- A workspace, if you chose to create it.
- If you chose to register the desktop app group, the registration will be completed.
- Virtual machines, if you chose to create them, which are joined to the domain and registered with the new host pool.
- A download link for an Azure Resource Management template based on your configuration.

After that, you're all done!

### **Run the Azure Resource Manager template to provision a new host pool**

If you'd rather use an automated process, download our Azure Resource Manager template to provision your new host pool instead.

## **2. Manage app groups**

### **Create a RemoteApp group**

If you've already created a host pool and session host VMs using the Azure portal or PowerShell, you can add application groups from the Azure portal with the following process:

1. Sign in to the Azure portal.
2. Search for and select **Azure Virtual Desktop**.
3. You can add an application group directly or you can add it from an existing host pool. Choose an option below:
  - Select **Application groups** in the menu on the left side of the page, then select **+ Add**.
  - Select **Host pools** in the menu on the left side of the screen, select the name of the host pool, select **Application groups** from the menu on the left side, then select **+ Add**. In this case, the host pool will already be selected on the Basics tab.
4. On the **Basics** tab, select the **Subscription** and **Resource group** you want to create the app group for. You can also choose to create a new resource group instead of selecting an existing one.
5. Select the **Host pool** that will be associated with the application group from the drop-down menu.

**Create an application group**

Basics	Assignments	Applications	Workspace	Tags	Review + create
Subscription * ⓘ	Microsoft Azure				
Resource group * ⓘ	Select a resource group Create new				
Host pool * ⓘ	0224HP				
Location ⓘ	West US Metadata stored in same location as host pool				
<b>Application group type</b> RemoteApp application groups are where you can add applications. A Desktop application group will grant full desktop access.					
Application group type * ⓘ	<input checked="" type="radio"/> RemoteApp <input type="radio"/> Desktop				
Application group name *					

6. Select **RemoteApp** under **Application group type**, then enter a name for your RemoteApp.

#### Application group type

RemoteApp application groups are where you can add applications. A Desktop application group will grant full desktop access.

Application group type \*

RemoteApp  Desktop

Application group name \*

7. Select **Next: Assignments >** tab.
8. To assign individual users or user groups to the app group, select **+Add Azure AD users or user groups**.
9. Select the users you want to have access to the apps. You can select single or multiple users and user groups.

#### Select Azure AD users or user groups X

Select member or invite an external user  ⓘ

Search by name or email address ✓

A dropdown menu showing a list of Azure AD users and groups. The list includes:

- AC Accounts
- AD ADSyncAdmins
- AD ADSyncAdmins
- AD ADSyncBrowse

Selected members:

A list of selected members, showing the user icon, name, email, and remove button.

	RdsDemoUser1 RdsDemoUser1@RdsPTTent...	<a href="#">Remove</a>
	Accounts	<a href="#">Remove</a>

**Select**

10. Select **Select**.

11. Select **Next: Applications >**, then select **+Add applications**.

12. To add an application from the start menu:

- Under **Application source**, select **Start menu** from the drop-down menu.
- Next, under **Application**, choose the application from the drop-down menu.

**Add application** X

Select an application from your start menu or add from a file path.

<b>Application source *</b>	<b>Start menu</b>
<b>Application *</b>	<b>Character Map</b>
<b>Display name</b>	<b>Character Map</b>
<b>Description</b>	
<b>Application path</b>	C:\windows\system32\charmap.exe
<b>Icon path</b>	C:\windows\system32\charmap.exe
<b>Icon index</b>	0
<b>Show in web feed</b>	<input type="radio"/> No <input checked="" type="radio"/> Yes
<b>Require command line</b>	<input checked="" type="radio"/> No <input type="radio"/> Yes

---

Save Cancel

- In **Display name**, enter the name for the application that will be shown to the user on their client.
- Leave the other options as-is and select **Save**.

13. To add an application from a specific file path:

- Under **Application source**, select **File path** from the drop-down menu.
- In **Application path**, enter the path to the application on the session host registered with the associated host pool.
- Enter the application's details in the **Application name**, **Display name**, **Icon path**, and **Icon index** fields.
- Select **Save**.

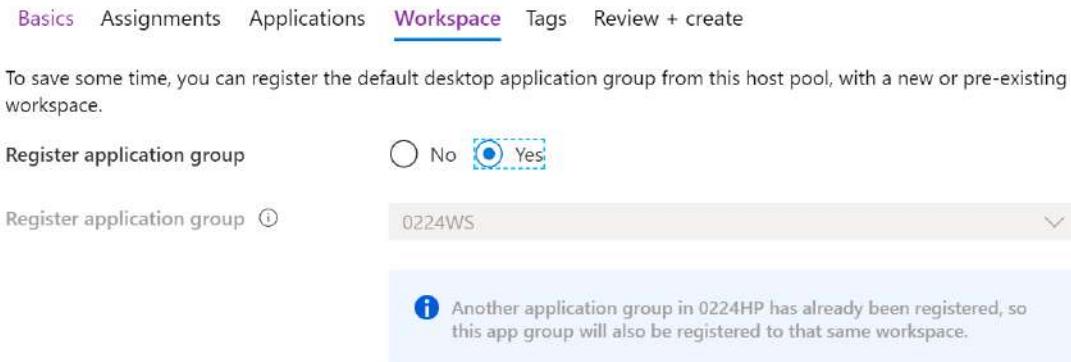
**Add application** X

Select an application from your start menu or add from a file path.

<b>Application source *</b> Application path * Application name * Display name Icon path * Icon index * Description	<input style="width: 150px; border: 1px solid #ccc; border-radius: 5px; padding: 2px 5px; margin-bottom: 5px;" type="button" value="File path"/> <div style="border: 1px solid #ccc; border-radius: 5px; padding: 2px 10px; width: 150px; height: 150px; background-color: #f0f0f0; margin-bottom: 5px;"></div> <input type="text" value="C:\windows\system32\charmap.exe"/> <input type="text" value="Character Map"/> <input type="text" value="false"/> <input type="text" value="C:\windows\system32\charmap.exe"/> <input type="text" value="0"/> <div style="border: 1px solid #ccc; border-radius: 5px; padding: 5px; min-height: 100px; margin-top: 10px;"></div>
Show in web feed <input type="radio"/> No <input checked="" type="radio"/> Yes	
Require command line <input checked="" type="radio"/> No <input type="radio"/> Yes	

---

14. Repeat this process for every application you want to add to the application group.
15. Next, select **Next: Workspace >**.
16. If you want to register the app group to a workspace, select **Yes** for **Register application group**. If you'd rather register the app group at a later time, select **No**.
17. If you select **Yes**, you can select an existing workspace to register your app group to.



18. Optionally, if you want to create tags to make your workspace easy to organize, select **Next: Tags >** and enter your tag names.
19. When you're done, select **Review + create**.
20. Wait a bit for the validation process to complete. When it's done, select **Create** to deploy your app group.

The deployment process will do the following things for you:

- Create the RemoteApp app group.
- Add your selected apps to the app group.
- Publish the app group published to users and user groups you selected.
- Register the app group, if you chose to do so.
- Create a link to an Azure Resource Manager template based on your configuration that you can download and save for later.

### Edit or remove an app

To edit or remove an app from an app group:

1. Sign in to the Azure portal.
2. Search for and select **Azure Virtual Desktop**.

3. You can either add an application group directly or from an existing host pool by choosing one of the following options:
  - To add a new application group directly, select **Application groups** in the menu on the left side of the page, then select the app group you want to edit.
  - To edit an app group in an existing host pool, select **Host pools** in the menu on the left side of the screen, select the name of the host pool, then select **Application groups** in the menu that appears on the left side of the screen, and then select the app group you want to edit.
4. Select **Applications** in the menu on the left side of the page.
5. If you want to remove an application, select the check box next to the application, then select **Remove** from the menu on the top of the page.
6. If you want to edit the details of an application, select the application name. This will open up the editing menu.
7. When you're done making changes, select **Save**.

### **3. Create a host pool to validate service updates**

Host pools are a collection of one or more identical virtual machines within Azure Virtual Desktop environment. We highly recommend you create a validation host pool where service updates are applied first. Validation host pools let you monitor service updates before the service applies them to your standard or non-validation environment. Without a validation host pool, you may not discover changes that introduce errors, which could result in downtime for users in your standard environment.

To ensure your apps work with the latest updates, the validation host pool should be as similar to host pools in your non-validation environment as possible. Users should connect as frequently to the validation host pool as they do to the standard host pool. If you have automated testing on your host pool, you should include automated testing on the validation host pool.

You can debug issues in the validation host pool with either the diagnostics feature or the Azure Virtual Desktop troubleshooting articles.

#### **Create your host pool**

You can configure any existing pooled or personal host pool to be a validation host pool. You can also create a new host pool to use for validation by following the instructions in any of these articles:

#### **Define your host pool as a validation host pool**

To use the Azure portal to configure your validation host pool:

1. Sign in to the Azure portal at <https://portal.azure.com>.
2. Search for and select **Azure Virtual Desktop**.
3. In the Azure Virtual Desktop page, select **Host pools**.
4. Select the name of the host pool you want to edit.
5. Select **Properties**.
6. In the validation environment field, select **Yes** to enable the validation environment.
7. Select **Save** to apply the new settings.

### **Update schedule**

Service updates happen monthly. If there are major issues, critical updates will be provided at a more frequent pace.

If there are any service updates, make sure you have at least a couple of users sign in each day to validate the environment. We recommend you regularly visit our TechCommunity site and follow any posts with WVDUPdate to stay informed about service updates.

## **4. Set up service alerts**

You can use Azure Service Health to monitor service issues and health advisories for Azure Virtual Desktop. Azure Service Health can notify you with different types of alerts (for example, email or SMS), help you understand the effect of an issue, and keep you updated as the issue resolves. Azure Service Health can also help you mitigate downtime and prepare for planned maintenance and changes that could affect the availability of your resources.

### **Create service alerts**

This shows you how to configure Azure Service Health and how to set up notifications, which you can access on the Azure portal. You can set up different types of alerts and schedule them to notify you in a timely manner.

### **Recommended service alerts**

We recommend you create service alerts for the following health event types:

- **Service issue:** Receive notifications on major issues that impact connectivity of your users with the service or with the ability to manage your Azure Virtual Desktop tenant.

- **Health advisory:** Receive notifications that require your attention. The following are some examples of this type of notification:

- Virtual Machines (VMs) not securely configured as open port 3389
- Deprecation of functionality

### Configure service alerts

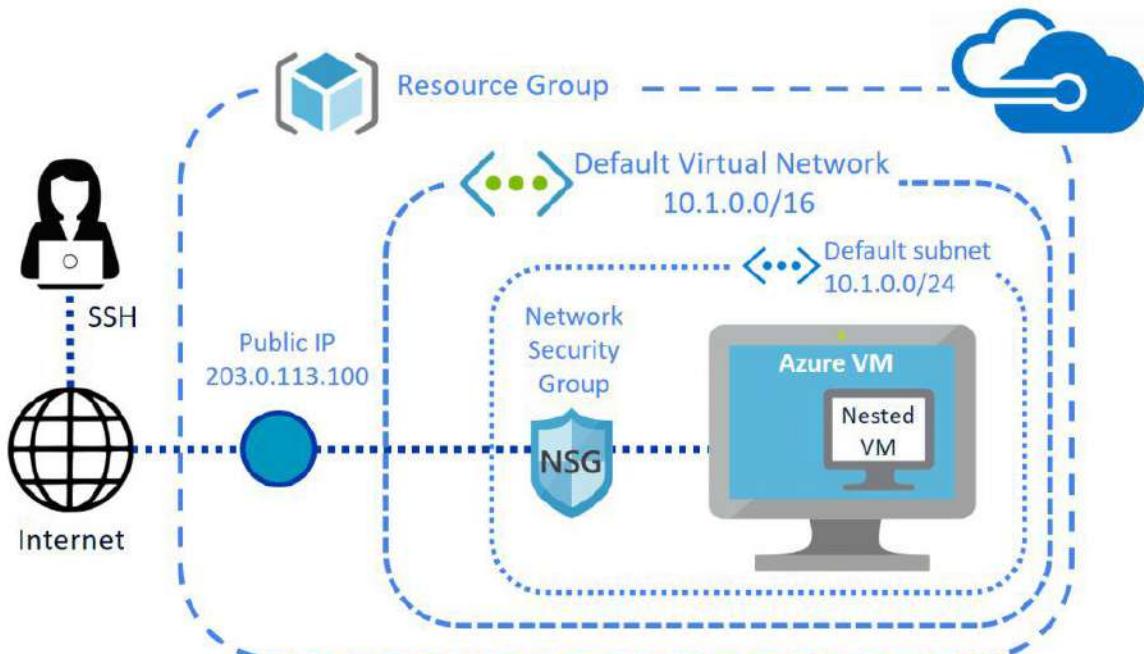
To configure service alerts:

1. Sign in to the Azure portal.
2. Select **Service Health**.
3. Follow the instructions in Create activity log alerts on service notifications to set up your alerts and notifications.

## VIRTUAL MACHINES

CloudSimple allows you to manage VMware virtual machines (VMs) from the Azure portal. A cluster or a resource pool from your vSphere cluster is managed through Azure by mapping it to your subscription.

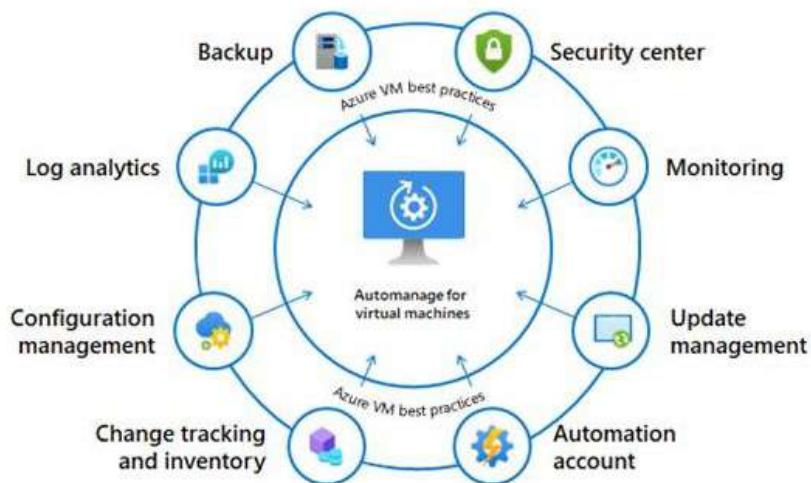
To create a CloudSimple VM from Azure, a VM template must exist on your Private Cloud vCenter. The template is used to customize the operating system and applications. The template VM can be hardened to meet enterprise security policies. You can use the template to create VMs and then consume them from the Azure portal using a self-service model.



## Benefits

CloudSimple virtual machines from Azure portal provide a self-service mechanism for users to create and manage VMware virtual machines.

- Create a CloudSimple VM on your Private Cloud vCenter
- Manage VM properties
  - Add/remove disks
  - Add/remove NICs
- Power operations of your CloudSimple VM
  - Power on and power off
  - Reset VM
- Delete VM



## Windows virtual machines in Azure

Azure Virtual Machines (VM) is one of several types of on-demand, scalable computing resources that Azure offers. Typically, you choose a VM when you need more control over the computing environment than the other choices offer. This article gives you information about what you should consider before you create a VM, how you create it, and how you manage it.

An Azure VM gives you the flexibility of virtualization without having to buy and maintain the physical hardware that runs it. However, you still need to maintain the VM by performing tasks, such as configuring, patching, and installing the software that runs on it.

Azure virtual machines can be used in various ways. Some examples are:

- **Development and test** – Azure VMs offer a quick and easy way to create a computer with specific configurations required to code and test an application.
- **Applications in the cloud** – Because demand for your application can fluctuate, it might make economic sense to run it on a VM in Azure. You pay for extra VMs when you need them and shut them down when you don't.
- **Extended datacenter** – Virtual machines in an Azure virtual network can easily be connected to your organization's network.

The number of VMs that your application uses can scale up and out to whatever is required to meet your needs.

## What are the prerequisites before creating a VM?

There are always a multitude of design considerations when you build out an application infrastructure in Azure. These aspects of a VM are important to think about before you start:

- The names of your application resources
- The location where the resources are stored
- The size of the VM
- The maximum number of VMs that can be created
- The operating system that the VM runs
- The configuration of the VM after it starts
- The related resources that the VM needs

## Locations

All resources created in Azure are distributed across multiple geographical regions around the world. Usually, the region is called **location** when you create a VM. For a VM, the location specifies where the virtual hard disks are stored.

This table shows some of the ways you can get a list of available locations.

Method	Description
Azure portal	Select a location from the list when you create a VM.
Azure PowerShell	Use the <a href="#">Get-AzLocation</a> command.
REST API	Use the <a href="#">List locations</a> operation.
Azure CLI	Use the <a href="#">az account list-locations</a> operation.

## Availability

Azure announced an industry leading single instance virtual machine Service Level Agreement of 99.9% provided you deploy the VM with premium storage for all disks. In order for your deployment to qualify for the standard 99.95% VM Service Level Agreement, you still need to deploy two or more VMs running your workload inside of an availability set. An availability set ensures that your VMs are distributed across multiple fault domains in the Azure data centers as well as deployed onto hosts with different maintenance windows. The full Azure SLA explains the guaranteed availability of Azure as a whole.

## VM size

The size of the VM that you use is determined by the workload that you want to run. The size that you choose then determines factors such as processing power, memory, and storage capacity. Azure offers a wide variety of sizes to support many types of uses.

Azure charges an hourly price based on the VM's size and operating system. For partial hours, Azure charges only for the minutes used. Storage is priced and charged separately.

## VM Limits

Your subscription has default quota limits in place that could impact the deployment of many VMs for your project. The current limit on a per subscription basis is 20 VMs per region. Limits can be raised by filing a support ticket requesting an increase.

## Operating system disks and images

Virtual machines use virtual hard disks (VHDs) to store their operating system (OS) and data. VHDs are also used for the images you can choose from to install an OS.

Azure provides many marketplace images to use with various versions and types of Windows Server operating systems. Marketplace images are identified by image publisher, offer, sku, and version (typically version is specified as latest). Only 64-bit operating systems are supported. For more information on the supported guest operating systems, roles, and features, see Microsoft server software support for Microsoft Azure virtual machines.

This table shows some ways that you can find the information for an image.

Method	Description
Azure portal	The values are automatically specified for you when you select an image to use.
Azure PowerShell	<a href="#">Get-AzVMImagePublisher</a> -Location <i>location</i> <a href="#">Get-AzVMImageOffer</a> -Location <i>location</i> -Publisher <i>publisherName</i> <a href="#">Get-AzVMImageSku</a> -Location <i>location</i> -Publisher <i>publisherName</i> - Offer <i>offerName</i>
REST APIs	<a href="#">List image publishers</a> <a href="#">List image offers</a> <a href="#">List image skus</a>
Azure CLI	<a href="#">az vm image list-publishers</a> --location <i>location</i> <a href="#">az vm image list-offers</a> --location <i>location</i> --publisher <i>publisherName</i> <a href="#">az vm image list-skus</a> --location <i>location</i> --publisher <i>publisherName</i> --offer <i>offerName</i>

## VM SCALE SETS

Azure virtual machine scale sets let you create and manage a group of load balanced VMs. The number of VM instances can automatically increase or decrease in response to demand or a defined schedule. Scale sets provide high availability to your applications, and allow you to centrally manage, configure, and update a large number of VMs. With virtual machine scale sets, you can build large-scale services for areas such as compute, big data, and container workloads.

### Benefits of VM Scale Sets

To provide redundancy and improved performance, applications are typically distributed across multiple instances. Customers may access your application through a load

balancer that distributes requests to one of the application instances. If you need to perform maintenance or update an application instance, your customers must be distributed to another available application instance. To keep up with extra customer demand, you may need to increase the number of application instances that run your application.

Azure virtual machine scale sets provide the management capabilities for applications that run across many VMs, automatic scaling of resources, and load balancing of traffic. Scale sets provide the following key benefits:

- **Easy to create and manage multiple VMs**
  - When you have many VMs that run your application, it's important to maintain a consistent configuration across your environment. For reliable performance of your application, the VM size, disk configuration, and application installs should match across all VMs.
  - With scale sets, all VM instances are created from the same base OS image and configuration. This approach lets you easily manage hundreds of VMs without extra configuration tasks or network management.
  - Scale sets support the use of the Azure load balancer for basic layer-4 traffic distribution, and Azure Application Gateway for more advanced layer-7 traffic distribution and TLS termination.
- **Provides high availability and application resiliency**
  - Scale sets are used to run multiple instances of your application. If one of these VM instances has a problem, customers continue to access your application through one of the other VM instances with minimal interruption.
  - For more availability, you can use Availability Zones to automatically distribute VM instances in a scale set within a single datacenter or across multiple datacenters.
- **Allows your application to automatically scale as resource demand changes**
  - Customer demand for your application may change throughout the day or week. To match customer demand, scale sets can automatically increase the number of VM instances as application demand increases, then reduce the number of VM instances as demand decreases.
  - Autoscale also minimizes the number of unnecessary VM instances that run your application when demand is low, while customers continue to receive an acceptable level of performance as demand grows and additional VM instances are automatically added. This ability helps reduce costs and efficiently create Azure resources as required.

- **Works at large-scale**

- Scale sets support up to 1,000 VM instances for standard marketplace images and custom images through the Azure Compute Gallery. If you create a scale set using a managed image, the limit is 600 VM instances.
- For the best performance with production workloads, use Azure Managed Disks.

## Differences between virtual machines and scale sets

Scale sets are built from virtual machines. With scale sets, the management and automation layers are provided to run and scale your applications. You could instead manually create and manage individual VMs, or integrate existing tools to build a similar level of automation. The following table outlines the benefits of scale sets compared to manually managing multiple VM instances.

Scenario	Manual group of VMs	Virtual machine scale set
Add extra VM instances	Manual process to create, configure, and ensure compliance	Automatically create from central configuration
Traffic balancing and distribution	Manual process to create and configure Azure load balancer or Application Gateway	Can automatically create and integrate with Azure load balancer or Application Gateway
High availability and redundancy	Manually create Availability Set or distribute and track VMs across Availability Zones	Automatic distribution of VM instances across Availability Zones or Availability Sets
Scaling of VMs	Manual monitoring and Azure Automation	Autoscale based on host metrics, in-guest metrics, Application Insights, or schedule

There is no extra cost to scale sets. You only pay for the underlying compute resources such as the VM instances, load balancer, or Managed Disk storage. The management and automation features, such as autoscale and redundancy, incur no additional charges over the use of VMs.

## **How to monitor your scale sets**

Use Azure Monitor for VMs, which has a simple onboarding process and will automate the collection of important CPU, memory, disk, and network performance counters from the VMs in your scale set. It also includes extra monitoring capabilities and pre-defined visualizations that help you focus on the availability and performance of your scale sets.

Enable monitoring for your virtual machine scale set application with Application Insights to collect detailed information about your application including page views, application requests, and exceptions. Further verify the availability of your application by configuring an availability test to simulate user traffic.

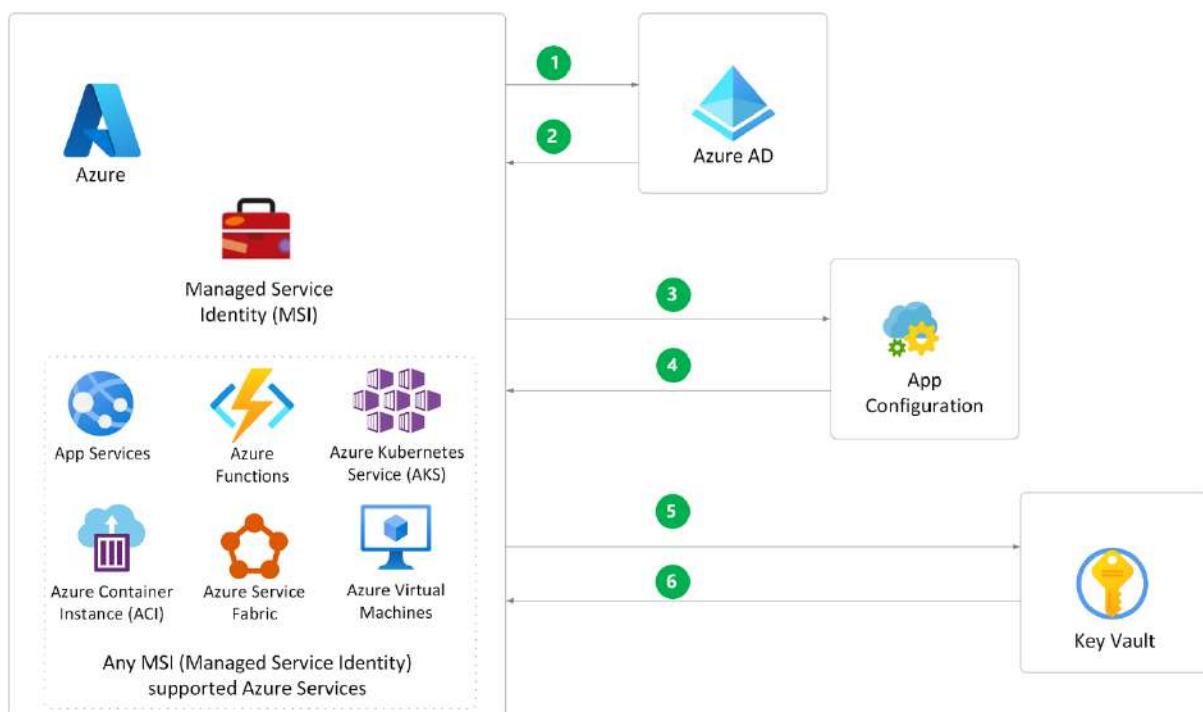
## **Data residency**

In Azure, the feature to enable storing customer data in a single region is currently only available in the Southeast Asia Region (Singapore) of the Asia Pacific Geo and Brazil South (Sao Paulo State) Region of Brazil Geo. Customer data is stored in Geo for all other regions. See Trust Center for more information.

# AZURE DEVELOPMENT

## APP CONFIGURATION

Azure App Configuration provides a service to centrally manage application settings and feature flags. Modern programs, especially programs running in a cloud, generally have many components that are distributed in nature. Spreading configuration settings across these components can lead to hard-to-troubleshoot errors during an application deployment. Use App Configuration to store all the settings for your application and secure their accesses in one place.



## Features

Cloud-based applications often run on multiple virtual machines or containers in multiple regions and use multiple external services. Creating a robust and scalable application in a distributed environment presents a significant challenge.

Various programming methodologies help developers deal with the increasing complexity of building applications. For example, the Twelve-Factor App describes many well-tested architectural patterns and best practices for use with cloud applications. One key recommendation from this guide is to separate configuration from code. An application's configuration settings should be kept external to its executable and read in from its runtime environment or an external source.

While any application can make use of App Configuration, the following examples are the types of application that benefit from the use of it:

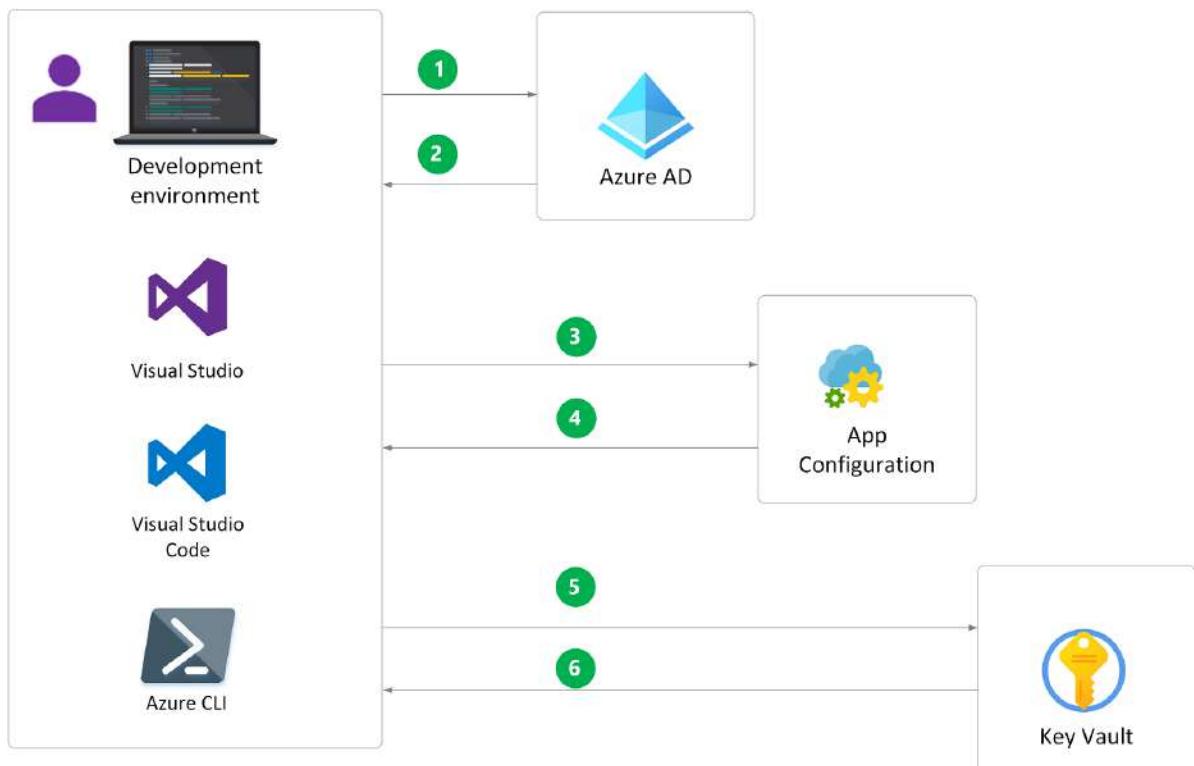
- Microservices based on Azure Kubernetes Service, Azure Service Fabric, or other containerized apps deployed in one or more geographies
- Serverless apps, which include Azure Functions or other event-driven stateless compute apps
- Continuous deployment pipeline

App Configuration offers the following benefits:

- A fully managed service that can be set up in minutes
- Flexible key representations and mappings
- Tagging with labels
- Point-in-time replay of settings
- Dedicated UI for feature flag management
- Comparison of two sets of configurations on custom-defined dimensions
- Enhanced security through Azure-managed identities
- Encryption of sensitive information at rest and in transit
- Native integration with popular frameworks

App Configuration complements Azure Key Vault, which is used to store application secrets. App Configuration makes it easier to implement the following scenarios:

- Centralize management and distribution of hierarchical configuration data for different environments and geographies
- Dynamically change application settings without the need to redeploy or restart an application
- Control feature availability in real-time



## Use App Configuration

The easiest way to add an App Configuration store to your application is through a client library provided by Microsoft. The following methods are available to connect with your application, depending on your chosen language and framework

Programming language and framework	How to connect
.NET Core and ASP.NET Core	App Configuration provider for .NET Core
.NET Framework and ASP.NET	App Configuration builder for .NET
Java Spring	App Configuration client for Spring Cloud
Others	App Configuration REST API

## Azure App Configuration best practices

### Key groupings

App Configuration provides two options for organizing keys:

- Key prefixes
- Labels

You can use either one or both options to group your keys.

*Key prefixes* are the beginning parts of keys. You can logically group a set of keys by using the same prefix in their names. Prefixes can contain multiple components connected by a delimiter, such as /, similar to a URL path, to form a namespace. Such hierarchies are useful when you're storing keys for many applications and microservices in one App Configuration store.

An important thing to keep in mind is that keys are what your application code references to retrieve the values of the corresponding settings. Keys shouldn't change, or else you'll have to modify your code each time that happens.

*Labels* are an attribute on keys. They're used to create variants of a key. For example, you can assign labels to multiple versions of a key. A version might be an iteration, an environment, or some other contextual information. Your application can request an entirely different set of key values by specifying another label. As a result, all key references remain unchanged in your code.

### Key-value compositions

App Configuration treats all keys stored with it as independent entities. App Configuration doesn't attempt to infer any relationship between keys or to inherit key values based on their hierarchy. You can aggregate multiple sets of keys, however, by using labels coupled with proper configuration stacking in your application code.

Let's look at an example. Suppose you have a setting named **Asset1**, whose value might vary based on the development environment. You create a key named "Asset1" with an empty label and a label named "Development". In the first label, you put the default value for **Asset1**, and you put a specific value for "Development" in the latter.

In your code, you first retrieve the key values without any labels, and then you retrieve the same set of key values a second time with the "Development" label. When you retrieve the values the second time, the previous values of the keys are overwritten. The .NET Core configuration system allows you to "stack" multiple sets of configuration data on top of each other. If a key exists in more than one set, the last set that contains it is used. With a

modern programming framework, such as .NET Core, you get this stacking capability for free if you use a native configuration provider to access App Configuration. The following code snippet shows how you can implement stacking in a .NET Core application:

```
// Augment the ConfigurationBuilder with Azure App Configuration  
// Pull the connection string from an environment variable  
configBuilder.AddAzureAppConfiguration(options => {  
    options.Connect(configuration["connection_string"])  
        .Select(KeyFilter.Any, LabelFilter.Null)  
        .Select(KeyFilter.Any, "Development");  
});
```

## References to external data

App Configuration is designed to store any configuration data that you would normally save in configuration files or environment variables. However, some types of data may better suited to reside in other sources. For example, store secrets in Key Vault, files in Azure Storage, membership information in Azure AD groups, or customer lists in a database.

You can still take advantage of App Configuration by saving a reference to external data in a key-value. You can use content type to differentiate each data source. When your application reads a reference, you load the data from the referenced source. In case that you change the location of your external data, you only need to update the reference in App Configuration instead of updating and redeploying your entire application.

The App Configuration Key Vault reference feature is an example in this case. It allows the secrets required for an application to be updated as necessary while the underlying secrets themselves remain in Key Vault.

## App Configuration bootstrap

To access an App Configuration store, you can use its connection string, which is available in the Azure portal. Because connection strings contain credential information, they're considered secrets. These secrets need to be stored in Azure Key Vault, and your code must authenticate to Key Vault to retrieve them.

A better option is to use the managed identities feature in Azure Active Directory. With managed identities, you need only the App Configuration endpoint URL to bootstrap access to your App Configuration store. You can embed the URL in your application code (for example, in the *appsettings.json* file).

## App or function access to App Configuration

You can provide access to App Configuration for Web Apps or Azure Functions by using any of the following methods:

- Through the Azure portal, enter the connection string to your App Configuration store in the Application settings of App Service.
- Store the connection string to your App Configuration store in Key Vault and reference it from App Service.
- Use Azure managed identities to access the App Configuration store. For more information, see [Integrate with Azure managed identities](#).
- Push configuration from App Configuration to App Service. App Configuration provides an export function (in Azure portal and the Azure CLI) that sends data directly into App Service. With this method, you don't need to change the application code at all.

## Reduce requests made to App Configuration

Excessive requests to App Configuration can result in throttling or overage charges. To reduce the number of requests made:

- Increase the refresh timeout, especially if your configuration values do not change frequently. Specify a new refresh timeout using the `SetCacheExpiration` method.
- Watch a single *sentinel key*, rather than watching individual keys. Refresh all configuration only if the sentinel key changes.
- Use Azure Event Grid to receive notifications when configuration changes, rather than constantly polling for any changes.
- Spread your requests across multiple App Configuration stores. For example, use a different store from each geographic region for a globally deployed application. Each App Configuration store has its own request quota. This setup gives you a model for scalability and avoids the single point of failure.

## Importing configuration data into App Configuration

App Configuration offers the option to bulk import your configuration settings from your current configuration files using either the Azure portal or CLI. You can also use the same options to export key-values from App Configuration, for example between related stores. If you'd like to set up an ongoing sync with your repo in GitHub or Azure DevOps,

you can use our GitHub Action or Azure Pipeline Push Task so that you can continue using your existing source control practices while getting the benefits of App Configuration.

## **Multi-region deployment in App Configuration**

App Configuration is regional service. For applications with different configurations per region, storing these configurations in one instance can create a single point of failure. Deploying one App Configuration instances per region across multiple regions may be a better option. It can help with regional disaster recovery, performance, and security siloing. Configuring by region also improves latency and uses separated throttling quotas, since throttling is per instance. To apply disaster recovery mitigation, you can use multiple configuration stores.

## **Client applications in App Configuration**

When you use App Configuration in client applications, ensure that you consider two major factors. First, if you're using the connection string in a client application, you risk exposing the access key of your App Configuration store to the public. Second, the typical scale of a client application might cause excessive requests to your App Configuration store, which can result in overage charges or throttling.

To address these concerns, we recommend that you use a proxy service between your client applications and your App Configuration store. The proxy service can securely authenticate with your App Configuration store without a security issue of leaking authentication information. You can build a proxy service by using one of the App Configuration provider libraries, so you can take advantage of built-in caching and refresh capabilities for optimizing the volume of requests sent to App Configuration. The proxy service serves the configuration from its cache to your client applications, and you avoid the two potential issues that are discussed in this section.

## **Configuration as Code**

Configuration as code is a practice of managing configuration files under your source control system, for example, a git repository. It gives you benefits like traceability and approval process for any configuration changes. If you adopt configuration as code, App Configuration has tools to assist you in deploying your configuration data. This way, your applications can access the latest data from your App Configuration store(s).

- For GitHub, you can enable the App Configuration Sync GitHub Action for your repository. Changes to configuration files are synchronized to App Configuration automatically whenever a pull request is merged.
- For Azure DevOps, you can include the Azure App Configuration Push, an Azure pipeline task, in your build or release pipelines for data synchronization.
- You can also import configuration files to App Configuration using Azure CLI as part of your CI/CD system. For more information, see `az appconfig kv import`.

This model allows you to include validation and testing steps before committing data to App Configuration. If you use multiple App Configuration stores, you can also push the configuration data to them incrementally or all at once.

## Manage feature flags in Azure App Configuration

### Create feature flags

The Feature Manager in the Azure portal for App Configuration provides a UI for creating and managing the feature flags that you use in your applications.

To add a new feature flag:

1. Select **Feature Manager** > **+Add** to add a feature flag.

The screenshot shows the Azure App Configuration Feature Manager interface. The left sidebar contains navigation links: Overview, Activity log, Access control (IAM), Tags, Diagnose and solve problems, Configuration Explorer, Feature Manager (selected), Compare, Import/Export, Events, Settings, Access Keys, Properties, Locks, and Export template. The main area has a search bar, an 'Add' button, and a 'Refresh' button. A message says 'Browse' and 'Total 2 feature flags loaded'. A table lists the feature flags:

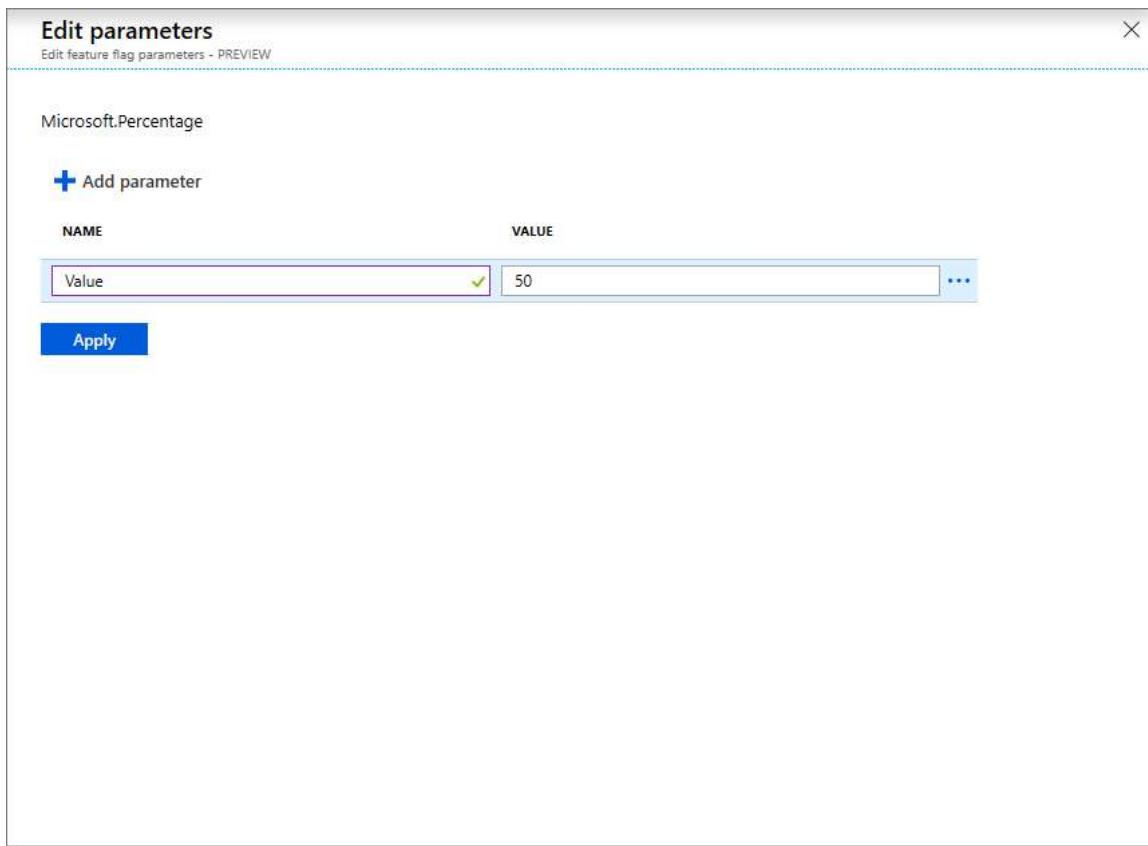
KEY	STATE	DESCRIPTION	LAST MODI...
FeatureA	Off <input type="button" value="On"/>		5/3/2019, ... <input type="button" value="..."/>
FeatureB	Off <input type="button" value="On"/>		5/3/2019, ... <input type="button" value="..."/>

2. Enter a unique key name for the feature flag. You need this name to reference the flag in your code.
3. If you want, give the feature flag a description.
4. Set the initial state of the feature flag. This state is usually *Off* or *On*. The *On* state changes to *Conditional* if you add a filter to the feature flag.

The screenshot shows the 'Add new feature flag - PREVIEW' dialog. The 'Conditional' radio button is selected. The 'Key' field contains 'FeatureC'. The 'Description' field is empty. A 'KEY' section shows 'Microsoft.Percentage' selected. At the bottom is an 'Apply' button.

5. When the state is *On*, select **+Add filter** to specify any additional conditions to qualify the state. Enter a built-in or custom filter key, and then select **+Add parameter** to associate one or more parameters with the filter. Built-in filters include:

Key	JSON parameters
Microsoft.Percentage	{"Value": 0-100 percent}
Microsoft.TimeWindow	{"Start": UTC time, "End": UTC time}
Microsoft.Targeting	{ "Audience": JSON blob defining users, groups, and rollout percentages.}



## Update feature flag states

To change a feature flag's state value:

1. Select **Feature Manager**.
2. To the right of a feature flag you want to modify, select the ellipsis (...), and then select **Edit**.
3. Set a new state for the feature flag.

## Access feature flags

Feature flags created by the Feature Manager are stored and retrieved as regular key values. They're kept under a special namespace prefix `.appconfig.featureflag`. To view the underlying key values, use the Configuration Explorer. Your application can retrieve these values by using the App Configuration configuration providers, SDKs, command-line extensions, and REST APIs.

## AZURE DEVOPS

Azure DevOps provides developer services for allowing teams to plan work, collaborate on code development, and build and deploy applications. Azure DevOps supports a culture and set of processes that bring developers, project managers, and contributors together to collaboratively develop software. It allows organizations to create and improve products at a faster pace than they can with traditional software development approaches.

You can work in the cloud using Azure DevOps Services or on-premises using Azure DevOps Server. Azure DevOps provides integrated features that you can access through your web browser or IDE client. You can use one or more of the following standalone services based on your business needs:

- **Azure Repos** provides Git repositories or Team Foundation Version Control (TFVC) for source control of your code.
- **Azure Pipelines** provides build and release services to support continuous integration and delivery of your applications.
- **Azure Boards** delivers a suite of Agile tools to support planning and tracking work, code defects, and issues using Kanban and Scrum methods.
- **Azure Test Plans** provides several tools to test your apps, including manual/exploratory testing and continuous testing.
- **Azure Artifacts** allows teams to share packages such as Maven, npm, NuGet, and more from public and private sources and integrate package sharing into your pipelines.

You can also use the following collaboration tools:

- Customizable team dashboards with configurable widgets to share information, progress, and trends
- Built-in wikis for sharing information
- Configurable notifications

Azure DevOps supports adding extensions and integrating with other popular services, such as: Campfire, Slack, Trello, UserVoice, and more, and developing your own custom extensions.

Azure DevOps Services supports integration with GitHub.com and GitHub Enterprise Server repositories. Azure DevOps Server supports integration with GitHub Enterprise Server repositories.

## Choose Azure DevOps Services

Choose Azure DevOps Services when you want the following outcomes:

- Quick set-up
- Maintenance-free operations
- Easy collaboration across domains
- Elastic scale
- Rock-solid security

Azure DevOps Services also gives you access to cloud build and deployment servers, and application insights.

You can use all the services included with Azure DevOps, or choose just what you need to complement your existing workflows.

- **Azure Boards.** Plan, track, and discuss work across your teams.
- **Azure Pipelines.** Continuously build, test, and deploy to any platform and cloud.
- **Azure Repos.** Get unlimited, cloud-hosted private Git repositories for your project.

## Choose Azure DevOps Server

Choose on-premises Azure DevOps Server when:

- You need your data to stay within your network.
- Your work tracking customization requirements are met better with the on-premises XML process model over the inheritance process model. The on-premises model supports modification of XML definition files.

When you deploy Azure DevOps Server, you can also configure the following servers or integration points:

- **Build server** supports on-premises and cloud-hosted builds.
- **SQL Server and SQL Analysis Server** support SQL Server Reports and the ability to create Excel pivot charts based on the cube.

## DevOps tools overview for Azure DevOps

Azure DevOps Services and Azure cloud services help remove barriers between teams, encourage collaboration, and improve the flow of value to your customers. Or, use our on-premises server, Team Foundation Server (TFS), when you want to maintain your data within your network.

Both options are enterprise-ready, supporting teams of any size, from tens to thousands. Azure DevOps Services provides a scalable, reliable, and globally available hosted service. It is backed by a 99.9% SLA, monitored by our 24-7 operations team, and available in local data centers around the world.

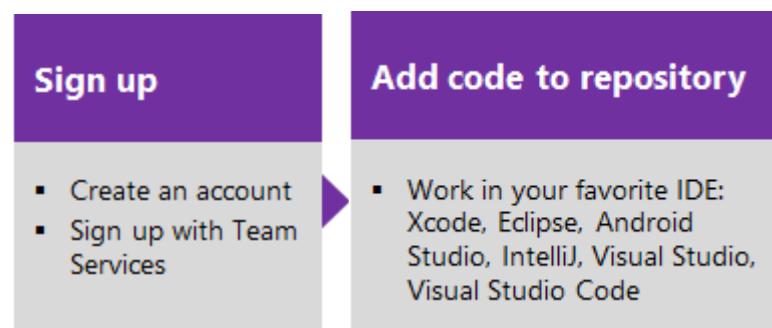
Also, you can quickly expand the power of these tools through integration with other services and tools using service hooks and extensions.

## Get started in the cloud or on-premises

Whether you work in the cloud, on-premises, or a hybrid of each, you have a comprehensive set of DevOps and Agile tools to support team collaboration throughout the cycles of planning, development and test, and continuous delivery.

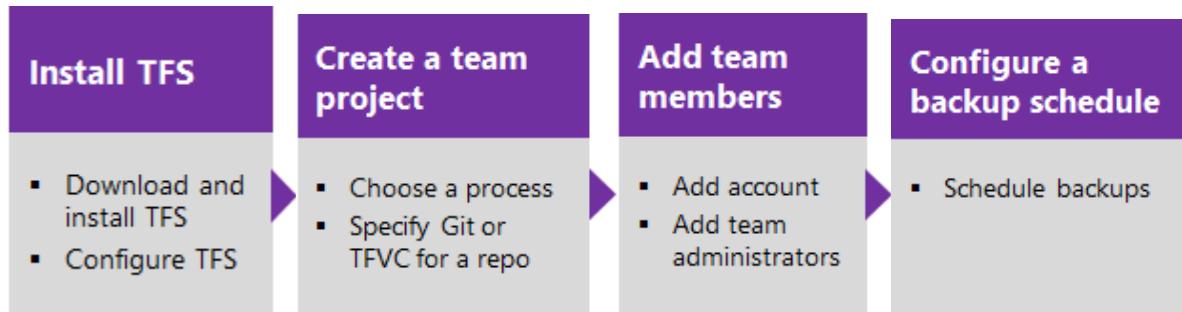
## Work in the cloud

Choose Azure DevOps Services when you want quick setup, maintenance-free operations, easy collaboration across domains, elastic scale, and rock solid security. You'll also have access to continuous testing, cloud build servers, and application insights. Small teams can start for free!



## Work on-premises

Choose on-premises when you need your data to stay within your network or you want access to SharePoint sites and SQL Server reporting services that integrate with data and tools.



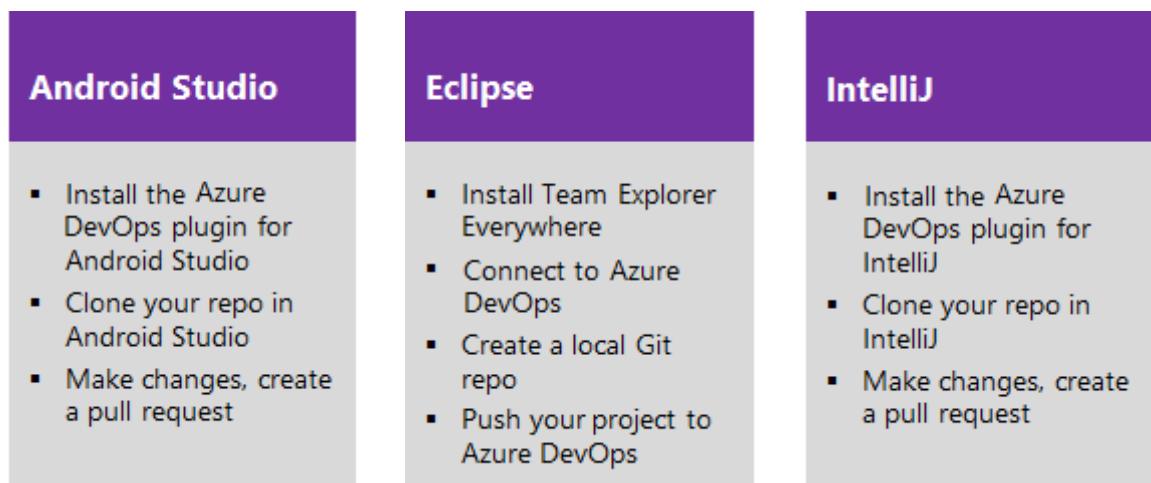
## Develop code using your IDE of choice

Azure DevOps Services supports two types of version control Git and Team Foundation Version Control (TFVC). Use Git, Team Foundation version control (TFVC) or both to store code for your app and give you access to different versions of your code.

Depending on whether you use Git or TFVC as a repo, you can develop your code in Android Studio, Eclipse, IntelliJ, Visual Studio, Visual Studio Code, or Xcode.

### Git

Git is a distributed version control system. Each developer has a copy of the source repository on their dev machine. Developers can commit each set of changes on their dev machine and perform version control operations such as history and compare without a network connection. Branches are lightweight.



Visual Studio	Visual Studio Code	Xcode
<ul style="list-style-type: none"> <li>▪ Connect to Azure DevOps</li> <li>▪ Create a local Git repo</li> <li>▪ Publish your code</li> <li>▪ Commit updates</li> <li>▪ Sync changes</li> </ul>	<ul style="list-style-type: none"> <li>▪ Connect to Azure DevOps</li> <li>▪ Create a local Git repo</li> <li>▪ Publish your code</li> <li>▪ Commit updates</li> <li>▪ Sync changes</li> </ul>	<ul style="list-style-type: none"> <li>▪ Create a local Git repo</li> <li>▪ Create a new repo in Azure DevOps</li> <li>▪ Push your project to Azure DevOps</li> </ul>

## TFVC

TFVC is a centralized version control system that lets you apply granular permissions and restrict access down to a file level. Typically, team members have only one version of each file on their dev machines. Historical data is maintained only on the server. Branches are path-based and created on the server.

You can use TFVC to scale from small to large projects, and by using server work spaces, you can scale up to very large code bases with millions of files per branch and large binary files. And with compare and annotate you can identify the exact changes that they made.

Eclipse	Visual Studio	Xcode
<ul style="list-style-type: none"> <li>▪ Install Team Explorer Everywhere</li> <li>▪ Connect to Azure DevOps</li> <li>▪ Create a local Git repo</li> <li>▪ Push your project to Azure DevOps</li> </ul>	<ul style="list-style-type: none"> <li>▪ Connect to Azure DevOps</li> <li>▪ Configure your workspace</li> <li>▪ Add your code to Azure DevOps</li> </ul>	<ul style="list-style-type: none"> <li>▪ Connect to Azure DevOps</li> <li>▪ Configure your workspace</li> <li>▪ Add your code to Azure DevOps</li> </ul>

## Package management and code search

Software development teams often rely on re-using libraries or providing libraries for others to re-use. Package management supports code sharing as binary components across

organizations and within teams. With it, you can build projects to produce packages and update projects that consume updated packages. Our Azure Artifacts extension enables plugging in existing package management services you already use, such as local NuGet servers for IP protection, NuGet, MyGet, or Artifactory.

- Azure Artifacts overview
- Search across all your code

Code Search provides a comprehensive solution to all your code exploration and troubleshooting needs. From discovering examples of API implementation to searching for error text, Code Search offers a fast and powerful way to find code. Search across one or more projects, with ranking and rich search results to ensure you find what you need and can focus in to understand your code. Code Search lets you filter your results based on code types such as definitions, comments, and references; filter by path, file extension, or repository; and use logical operators such as AND, OR, NOT to refine your query and get the results you need.

Code Search also makes team collaboration easier and helps maximize developer efficiency. View history and annotations to see who last changed a line of code, and what they changed. Search locally within code files, and find references or definitions of Code Search matches, when debugging or exploring your code. Add your comments and then communicate the results to team members easily by sharing the query URL.

## Plan and track work with Agile tools

Use Agile tools to plan and track work using Scrum and Kanban processes or a mix of both. Scrum tools support defining and managing work within sprints, setting capacity, and tracking tasks. Kanban tools allow you to manage a continuous flow of work via an interactive sign board. In addition, configurable charts, dashboards, and reports help teams monitor and share progress.

## Backlog planning

- Create your backlog
- Prioritize your backlog
- Estimate work
- Assign work

## Scrum: Plan sprints

- Assign work to a sprint
- Set team capacity
- Define tasks
- Adjust plan to fit capacity

## Kanban

- Track work in progress
- Update status
- Reorder on the fly

## Dashboards

- Add dashboards
- Add widgets
- Configure widgets

You also gain access to a rich set of customization capabilities.

## DevOps: Build - Test - Release

Help your team continuously deliver software at a faster pace and with lower risk, while improving efficiency and collaboration between all teams that participate in release processes. Set up continuous integration builds for your app that run with every check in. Multi-platform build agents support Android, iOS, Java, .NET, and other applications. Easily provision test environments. Track when the quality is sufficient to release to the customer.

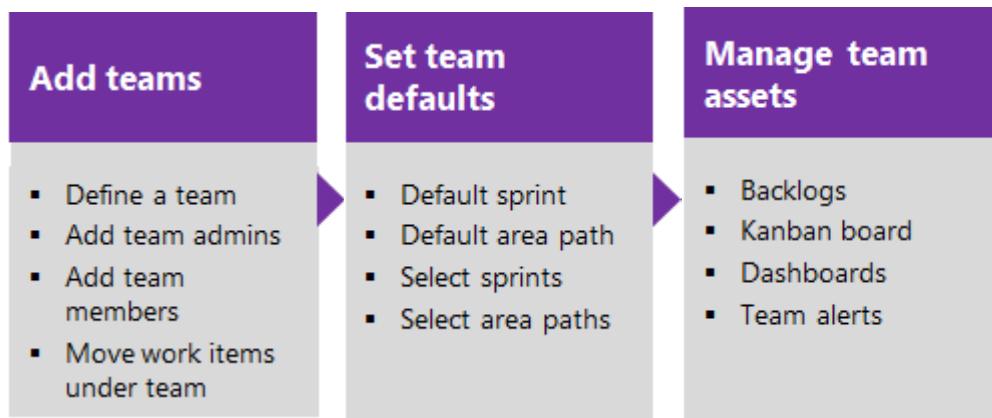


In addition to continuous integration testing, you can create test plans, perform manual testing, and run unit tests.

Azure Pipelines, and Build & Release in TFS, help you automate the deployment and testing of your software in multiple environments. With it, you can either fully automate the delivery of your software all the way to production, or set up semi-automated processes with approvals and on-demand deployments.

## Scale up

As your team grows, your tools grow. You can easily add teams which can focus on their set of backlog stories. Each team you create gets access to their set of dashboards, Agile planning tools, and other collaborative tools.



## Extensibility: Create first-class integration experiences

The extensibility framework enables you to build integrations directly within Azure DevOps to create first-class, seamless connections between different tools and services.

With Marketplace extensions (currently in private preview), you can create first-class integration experiences, such as a simple context menu or toolbar action. Or, you can create a complex, powerful full UI experience that seamlessly lights up within the Azure DevOps Services web portal.

- Find marketplace extensions
- Get extensions
- Using service hooks
- Get started with REST APIs

Service hooks enable integration scenarios between other applications and Azure DevOps by subscribing to events instead of constantly polling for them. Service hooks provide a more efficient way to drive activities when events happen in your projects. For example, you can send a push notification to your team's mobile devices when a build fails, or create a card in Trello when a work item is created. Some of the services you can easily integrate with are UserVoice, Zendesk, Trello, Slack, and HipChat.

Industry-standard RESTful APIs extend the power of Azure DevOps from your apps and services. With them, you can integrate from virtually any device, platform, or technology stack, including Android, iOS, Node.js, .NET, and more.

## AZURE SPRING CLOUD

Azure Spring Cloud makes it easy to deploy Spring Boot microservice applications to Azure without any code changes. The service manages the infrastructure of Spring Cloud applications so developers can focus on their code. Azure Spring Cloud provides lifecycle management using comprehensive monitoring and diagnostics, configuration management, service discovery, CI/CD integration, blue-green deployments, and more.

### Benefits

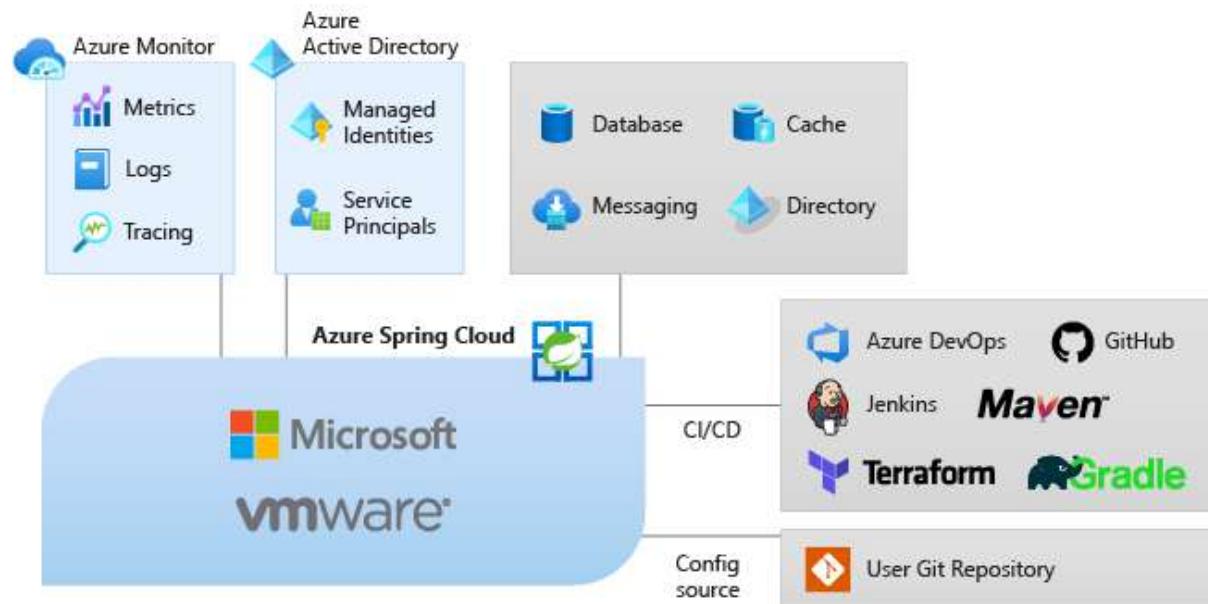
Deployment of applications to Azure Spring Cloud has many benefits. You can:

- Efficiently migrate existing Spring apps and manage cloud scaling and costs.
- Modernize apps with Spring Cloud patterns to improve agility and speed of delivery.
- Run Java at cloud scale and drive higher usage without complicated infrastructure.
- Develop and deploy rapidly without containerization dependencies.
- Monitor production workloads efficiently and effortlessly.

Azure Spring Cloud supports both Java Spring Boot and ASP.NET Core Steeltoe apps. Steeltoe support is currently offered as a public preview. Public preview offerings let you experiment with new features prior to their official release.

## Service overview

As part of the Azure ecosystem, Azure Spring Cloud allows easy binding to other Azure services including storage, databases, monitoring, and more.



- Azure Spring Cloud is a fully managed service for Spring Boot apps that lets you focus on building and running apps without the hassle of managing infrastructure.
- Simply deploy your JARs or code for your Spring Boot app or Zip for your Steeltoe app, and Azure Spring Cloud will automatically wire your apps with Spring service runtime and built-in app lifecycle.
- Monitoring is simple. After deployment you can monitor app performance, fix errors, and rapidly improve applications.
- Full integration to Azure's ecosystems and services.
- Azure Spring Cloud is enterprise ready with fully managed infrastructure, built-in lifecycle management, and ease of monitoring.

## Prepare an application for deployment in Azure Spring Cloud

### Java Runtime version

Azure Spring Cloud supports both Java 8 and Java 11. In general, Azure PaaS only supports Java LTS versions and Azure Spring Cloud will support Java 17 LTS. The hosting environment contains the latest version of Azul Zulu OpenJDK for Azure. For more information about Azul Zulu OpenJDK for Azure, see [Install the JDK](#).

### Spring Boot and Spring Cloud versions

To prepare an existing Spring Boot application for deployment to Azure Spring Cloud include the Spring Boot and Spring Cloud dependencies in the application POM file as shown in the following sections.

Azure Spring Cloud will support the latest Spring Boot or Spring Cloud release within one month after it's been released. You can get supported Spring Boot versions from [Spring Boot Releases](#) and Spring Cloud versions from [Spring Cloud Releases](#).

The following table lists the supported Spring Boot and Spring Cloud combinations:

Spring Boot version	Spring Cloud version
2.3.x	Hoxton.SR8+
2.4.x, 2.5.x	2020.0 aka Ilford +

### Dependencies for Spring Boot version 2.3

For Spring Boot version 2.3 add the following dependencies to the application POM file.

```
<!-- Spring Boot dependencies -->  
  
<parent>  
  
  <groupId>org.springframework.boot</groupId>  
  <artifactId>spring-boot-starter-parent</artifactId>  
  <version>2.3.4.RELEASE</version>  
  
</parent>
```

```
<!-- Spring Cloud dependencies -->  
  
<dependencyManagement>
```

```

<dependencies>
  <dependency>
    <groupId>org.springframework.cloud</groupId>
    <artifactId>spring-cloud-dependencies</artifactId>
    <version>Hoxton.SR8</version>
    <type>pom</type>
    <scope>import</scope>
  </dependency>
</dependencies>
</dependencyManagement>

```

## Dependencies for Spring Boot version 2.4/2.5

For Spring Boot version 2.4/2.5 add the following dependencies to the application POM file.

```

<!-- Spring Boot dependencies -->
<parent>
  <groupId>org.springframework.boot</groupId>
  <artifactId>spring-boot-starter-parent</artifactId>
  <version>2.4.8</version>
</parent>

<!-- Spring Cloud dependencies -->
<dependencyManagement>
  <dependencies>
    <dependency>
      <groupId>org.springframework.cloud</groupId>
      <artifactId>spring-cloud-dependencies</artifactId>
      <version>2020.0.2</version>
    </dependency>
  </dependencies>
</dependencyManagement>

```

```
<type>pom</type>
<scope>import</scope>
</dependency>
</dependencies>
</dependencyManagement>
```

## Service Registry

To use the managed Azure Service Registry service, include the spring-cloud-starter-netflix-eureka-client dependency in the pom.xml file as shown here:

```
<dependency>
  <groupId>org.springframework.cloud</groupId>
  <artifactId>spring-cloud-starter-netflix-eureka-client</artifactId>
</dependency>
```

The endpoint of the Service Registry server is automatically injected as environment variables with your app. Applications can register themselves with the Service Registry server and discover other dependent microservices.

## EnableDiscoveryClient annotation

Add the following annotation to the application source code.

```
@EnableDiscoveryClient
package com.piggymetrics.gateway;
```

```
import org.springframework.boot.SpringApplication;
import org.springframework.boot.autoconfigure.SpringBootApplication;
import org.springframework.cloud.client.discovery.EnableDiscoveryClient;
import org.springframework.cloud.netflix.zuul.EnableZuulProxy;
```

```
@SpringBootApplication  
 @EnableDiscoveryClient  
 @EnableZuulProxy  
  
public class GatewayApplication {  
    public static void main(String[] args) {  
        SpringApplication.run(GatewayApplication.class, args);  
    }  
}
```

## Distributed Configuration

To enable Distributed Configuration, include the following spring-cloud-config-client dependency in the dependencies section of your pom.xml file:

```
<dependency>  
    <groupId>org.springframework.cloud</groupId>  
    <artifactId>spring-cloud-config-client</artifactId>  
</dependency>  
  
<dependency>  
    <groupId>org.springframework.cloud</groupId>  
    <artifactId>spring-cloud-starter-bootstrap</artifactId>  
</dependency>
```

## Metrics

Include the spring-boot-starter-actuator dependency in the dependencies section of your pom.xml file as shown here:

```
<dependency>
```

```

<groupId>org.springframework.boot</groupId>
<artifactId>spring-boot-starter-actuator</artifactId>
</dependency>

```

## Deploy your first application to Azure Spring Cloud

### Generate a Spring Cloud project

Start with Spring Initializr to generate a sample project with recommended dependencies for Azure Spring Cloud. This link uses the following URL to provide default settings for you.

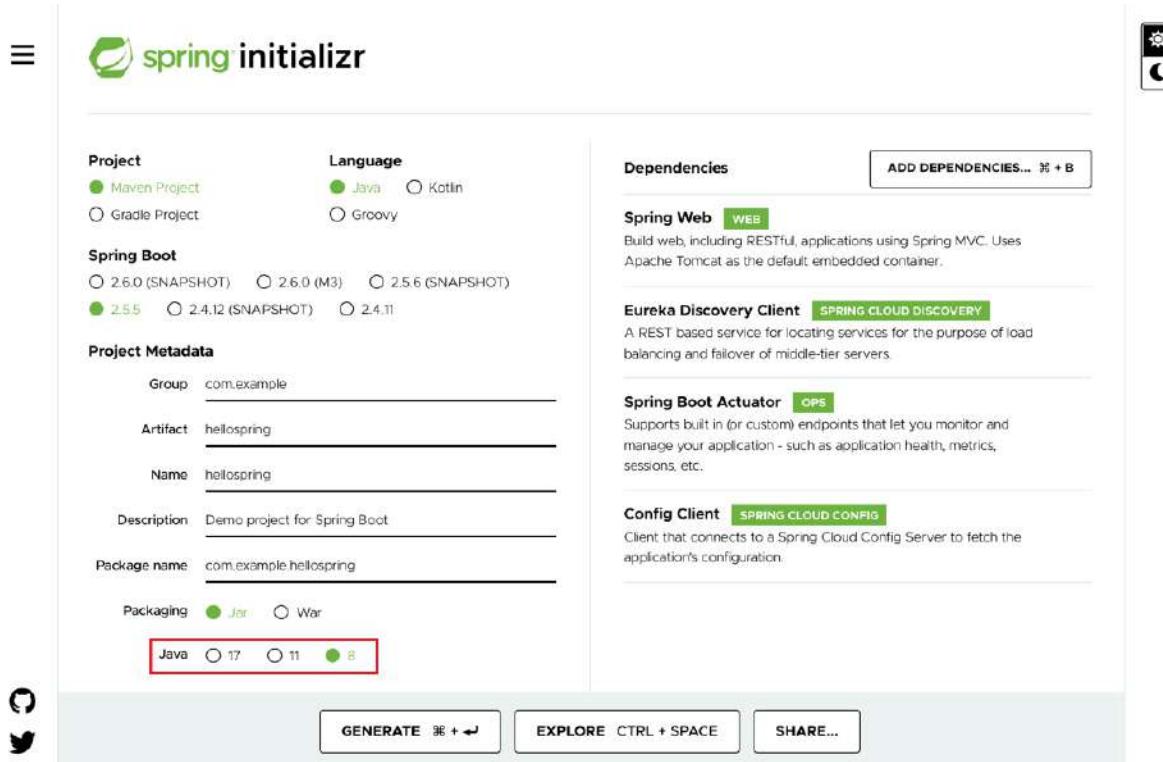
```

http://start.spring.io/#!type=maven-project&language=java&platformVersion=2.5.7&packaging=jar&jvmVersion=1.8&groupId=com.example&artifactId=hellospring&name=hellospring&description=Demo%20project%20for%20Spring%20Boot&packageName=com.example.hellospring&dependencies=web,cloud-eureka,actuator,cloud-config-client

```

The following image shows the recommended Initializr set up for this sample project.

This example uses Java version 8. If you want to use Java version 11, change the option under **Project Metadata**.



The screenshot shows the Spring Initializr web interface. On the left, there's a sidebar with a menu icon (three horizontal lines) and social sharing icons for LinkedIn and Twitter. The main area has a header with the Spring Initializr logo and a gear icon for settings. The configuration is as follows:

- Project:** Maven Project (selected)
- Language:** Java (selected)
- Spring Boot:** 2.5.5 (selected)
- Project Metadata:**
  - Group: com.example
  - Artifact: hellospring
  - Name: hellospring
  - Description: Demo project for Spring Boot
  - Package name: com.example.hellospring
  - Packaging: Jar (selected)
  - Java version: 8 (selected)
- Dependencies:**
  - Spring Web** (WEB): Build web, including RESTful, applications using Spring MVC. Uses Apache Tomcat as the default embedded container.
  - Eureka Discovery Client** (SPRING CLOUD DISCOVERY): A REST based service for locating services for the purpose of load balancing and failover of middle-tier servers.
  - Spring Boot Actuator** (OPS): Supports built in (or custom) endpoints that let you monitor and manage your application - such as application health, metrics, sessions, etc.
  - Config Client** (SPRING CLOUD CONFIG): Client that connects to a Spring Cloud Config Server to fetch the application's configuration.

At the bottom, there are buttons for **GENERATE** (⌘ + ↩), **EXPLORE** (CTRL + SPACE), and **SHARE...**.

1. Select **Generate** when all the dependencies are set.
2. Download and unpack the package, then create a web controller for a simple web application by adding the file `src/main/java/com/example/hellospring/HelloController.java` with the following contents:

```
package com.example.hellospring;

import org.springframework.web.bind.annotation.RestController;
import org.springframework.web.bind.annotation.RequestMapping;

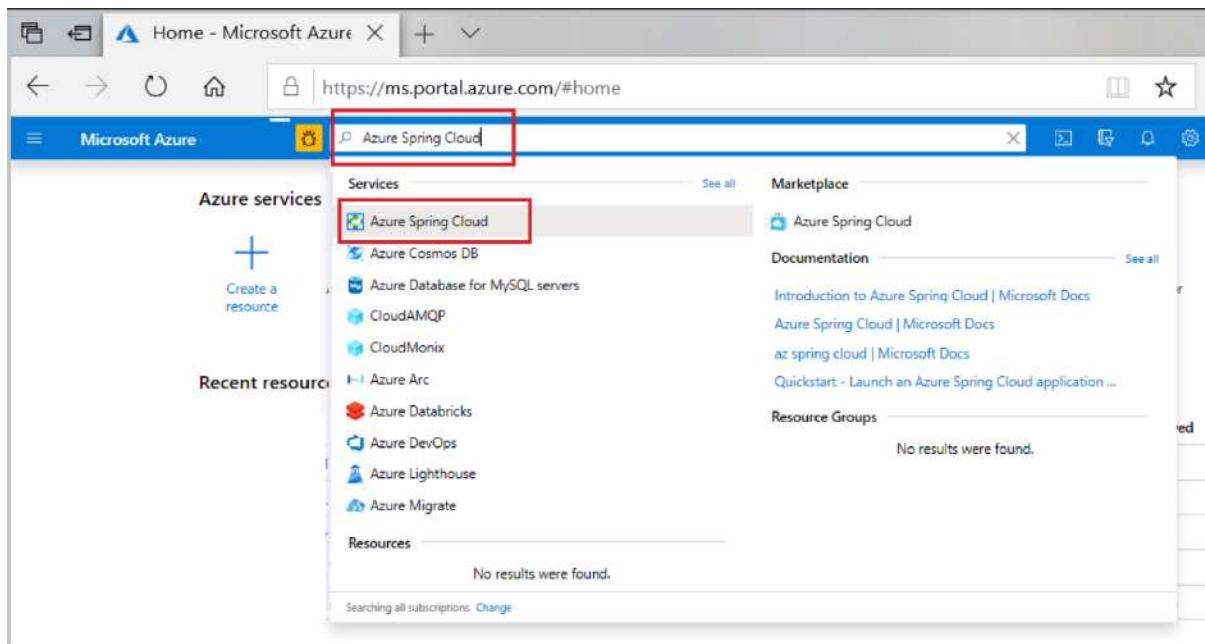
@RestController
public class HelloController {

    @RequestMapping("/")
    public String index() {
        return "Greetings from Azure Spring Cloud!";
    }
}
```

## Provision an instance of Azure Spring Cloud

The following procedure creates an instance of Azure Spring Cloud using the Azure portal.

1. In a new tab, open the Azure portal.
2. From the top search box, search for **Azure Spring Cloud**.
3. Select **Azure Spring Cloud** from the results.



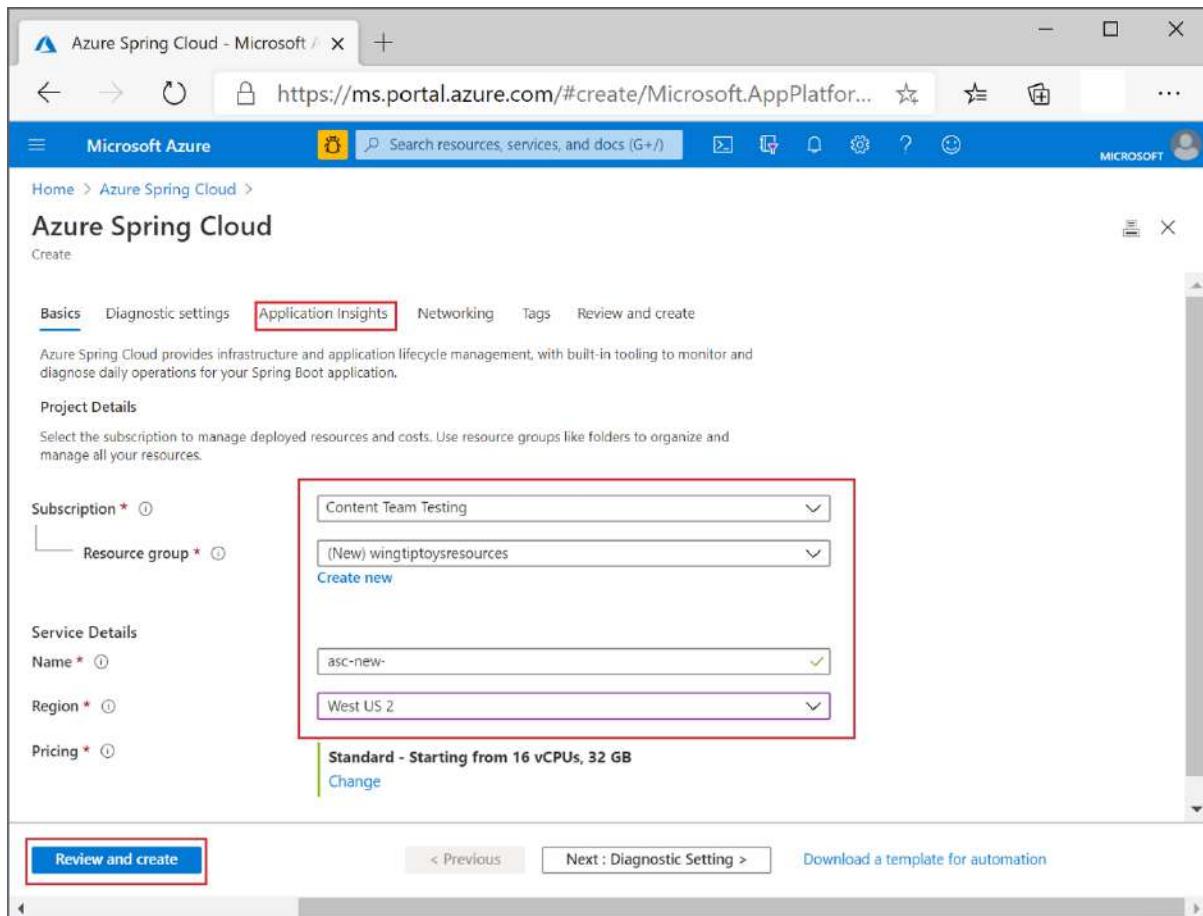
4. On the Azure Spring Cloud page, select **Create**.

A screenshot of the Azure Spring Cloud list page. The top navigation bar includes "Home &gt; Azure Spring Cloud" and the Microsoft logo. The main toolbar features a "Create" button (highlighted with a red box), "Manage view", "Refresh", "Export to CSV", "Open query", and "Feedback". Below the toolbar are filters for "Subscription == 35 of 46 selected" and "Resource group == all". The main content area displays a message "Showing 0 to 0 of 0 records." and a column header "Name ↑".

5. Fill out the form on the Azure Spring Cloud **Create** page. Consider the following guidelines:

- **Subscription:** Select the subscription you want to be billed for this resource.
- **Resource group:** Creating new resource groups for new resources is a best practice. You will use this resource group in later steps as **<resource group name>**.
- **Service Details/Name:** Specify the **<service instance name>**. The name must be between 4 and 32 characters long and can contain only lowercase letters, numbers, and hyphens. The first character of the service name must be a letter and the last character must be either a letter or a number.

- **Location:** Select the region for your service instance.



## 6. Select Review and create.

## Build and deploy the app

The following procedure builds and deploys the application using the Azure CLI. Execute the following command at the root of the project.

1. Sign in to Azure and choose your subscription.

```
az login
```

If you have more than one subscription, use the following command to list the subscriptions you have access to, then choose the one you want to use for this quickstart.

```
az account list -o table
```

Use the following command to set the default subscription to use with the Azure CLI commands in this quickstart.

```
az account set --subscription <Name or ID of a subscription from the last step>
```

## 2. Build the project using Maven:

```
mvn clean package -DskipTests
```

## 3. Create the app with a public endpoint assigned. If you selected Java version 11 when generating the Spring Cloud project, include the --runtime-version=Java\_11 switch.

```
az spring-cloud app create -n hellospring -s <service instance name> -g <resource group name> --assign-endpoint true
```

## 4. Deploy the Jar file for the app (target\hellospring-0.0.1-SNAPSHOT.jar on Windows):

```
az spring-cloud app deploy -n hellospring -s <service instance name> -g <resource group name> --artifact-path <jar file path>/hellospring-0.0.1-SNAPSHOT.jar
```

## 5. It takes a few minutes to finish deploying the application. To confirm that it has deployed, go to the Apps section in the Azure portal. You should see the status of the application.

Once deployment has completed, you can access the app at <https://<service instance name>-hellospring.azuremicroservices.io/>.



## Streaming logs in real time

Use the following command to get real-time logs from the App.

```
az spring-cloud app logs -n hellospring -s <service instance name> -g <resource group name> --lines 100 -f
```

Logs appear in the results:

```
PS C:\Users\user\Documents\vsworks\hellospring> az spring-cloud app logs -n hellospring -s userqs2 -g userqs2
Command group 'spring-cloud' is in preview. It may be changed/removed in a future release.
=====
:: Spring Boot ::      (v2.3.3.RELEASE)

2020-08-20 03:32:58.155  INFO [hellospring,,] 1 --- [k to default profiles: default
2020-08-20 03:33:04.030  WARN [hellospring,,] 1 --- [tains invalid characters, please migrate to a valid format.
2020-08-20 03:33:05.144  INFO [hellospring,,] 1 --- [
a04c-9e27571f6a0c
2020-08-20 03:33:08.492  INFO [hellospring,,] 1 --- [
025 (http)
2020-08-20 03:33:08.512  INFO [hellospring,,] 1 --- [
2020-08-20 03:33:08.513  INFO [hellospring,,] 1 --- [
tomcat/9.0.37]
2020-08-20 03:33:08.653  INFO [hellospring,,] 1 --- [
main] c.e.hellospring.HellospringApplication : No active profile set, falling bac
main] o.s.boot.actuate.endpoint.EndpointId : Endpoint ID 'service-registry' con
main] o.s.cloud.context.scope.GenericScope : BeanFactory id=767d548c-c478-3c99-
main] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat initialized with port(s): 1
main] o.apache.catalina.core.StandardService : Starting service [Tomcat]
main] org.apache.catalina.core.StandardEngine : Starting Servlet engine: [Apache T
main] o.a.c.c.C.[Tomcat].[localhost].[/] : Initializing Spring embedded WebAp
```

For advanced logs analytics features, visit the **Logs** tab in the menu on the Azure portal. Logs here have a latency of a few minutes.

TimeGenerated (UTC)	serviceName	appName	instanceName	Log
8/5/2020 7:30:17.474 AM	serviceName	demo	demo-default-4-59cc9b54d-vt856	2020-08-05 07:30:17.474 INFO [demo..] 1 --- [trap-executor-0] c.a.c
8/5/2020 7:45:17.476 AM	serviceName	demo	demo-default-4-59cc9b54d-vt856	2020-08-05 07:45:17.476 INFO [demo..] 1 --- [trap-executor-0] c.a.c
8/5/2020 7:50:17.476 AM	serviceName	demo	demo-default-4-59cc9b54d-vt856	2020-08-05 07:50:17.476 INFO [demo..] 1 --- [trap-executor-0] c.a.c
8/5/2020 7:55:17.477 AM	serviceName	demo	demo-default-4-59cc9b54d-vt856	2020-08-05 07:55:17.477 INFO [demo..] 1 --- [trap-executor-0] c.a.c
8/5/2020 7:55:17.471 AM	serviceName	demo	demo-default-4-59cc9b54d-vt856	2020-08-05 07:55:17.471 INFO [demo..] 1 --- [trap-executor-0] c.a.c
8/5/2020 7:55:17.472 AM	serviceName	demo	demo-default-4-59cc9b54d-vt856	2020-08-05 07:55:17.472 INFO [demo..] 1 --- [trap-executor-0] c.a.c
8/5/2020 7:55:17.479 AM	serviceName	demo	demo-default-4-59cc9b54d-vt856	2020-08-05 07:55:17.479 INFO [demo..] 1 --- [trap-executor-0] c.a.c
8/5/2020 7:55:17.473 AM	serviceName	demo	demo-default-4-59cc9b54d-vt856	2020-08-05 07:55:17.473 INFO [demo..] 1 --- [trap-executor-0] c.a.c

## Clean up resources

In the above steps, you created Azure resources that will continue to accrue charges while they remain in your subscription. If you don't expect to need these resources in the future, delete the resource group from the portal or by running the following command in the Azure CLI:

```
az group delete --name <your resource group name> --yes
```

## Azure Spring Cloud task

### Arguments

Argument	Action	Description
Action Action	All	(Required) The action to be performed by this task. <b>One of:</b> Deploy, Set Production, Delete Staging Deployment <b>Default value:</b> Deploy
ConnectedServiceName Azure Subscription	All	(Required) The name of the <a href="#">Azure Resource Manager service connection</a> . <b>Argument alias:</b> azureSubscription
AzureSpringCloud Azure Spring Cloud	All	(Required) The name or resource ID of the Azure Spring Cloud instance.

<b>Argument</b>	<b>Action</b>	<b>Description</b>
AppName App Name	All	(Required) The name of the Azure Spring Cloud app to deploy. The app must exist prior to task execution.
UseStagingDeployment Use Staging Deployment.	Deploy Set Production	(Required) If set to <code>true</code> , apply the task to whichever <code>deployment</code> is set as the staging deployment at time of execution. If omitted, the <code>DeploymentName</code> parameter must be set. Default value: <code>true</code>
DeploymentName Deployment Name	Deploy Set production	(Required if <code>UseStagingDeployment</code> is <code>false</code> ) The name of the <code>deployment</code> to which the action will apply. If not using blue-green deployments, set this field to <code>default</code> .
CreateNewDeployment Create new deployment	Deploy	(Optional) If set to <code>true</code> and the deployment specified by <code>DeploymentName</code> does not exist at execution time, it will be created. Default value: <code>false</code>
Package Package or folder	Deploy	(Required) The file path to the package containing the application to be deployed (.jar file for Java, .zip for .NET Core) or to a folder containing the application source to be built. <code>Build variables</code> or <code>release variables</code> are supported. Default value: <code>\$(System.DefaultWorkingDirectory)/**/*.jar</code>
RuntimeVersion Runtime Version	Deploy	(Optional) The runtime stack for the application. One of: <code>Java_8</code> , <code>Java_11</code> , <code>NetCore_31</code> , Default value: <code>Java_11</code>
EnvironmentVariables Environment Variables	Deploy	(Optional) Environment variables to be entered using the syntax '-key value'. Values containing spaces should be enclosed in double quotes. Example: <code>-CUSTOMER_NAME Contoso</code> <code>-WEBSITE_TIME_ZONE "Eastern Standard Time"</code>
JvmOptions JVM Options	Deploy	(Optional) A string containing JVM Options. Example: <code>-xx:+UseG1GC</code> <code>-XX:MaxRAMPercentage=70</code> <code>-Dazure.keyvault.enabled=true</code> <code>-Dazure.keyvault.uri=https://myvault.vault.azure.net/</code>

<b>Argument</b>	<b>Action</b>	<b>Description</b>
DotNetCoreMainEntryPath .NET Core entry path	Deploy	(Optional) A string containing the path to the .NET executable relative to zip root.
version Version	Deploy	(Optional) The deployment version. If not set, the version is left unchanged.

## Examples

### Deleting a staging deployment

The "Delete Staging Deployment" action allows you to delete the deployment not receiving production traffic. This frees up resources used by that deployment and makes room for a new staging deployment:

variables:

```
azureSubscription: Contoso
```

steps:

- task: AzureSpringCloud@0

    continueOnError: true # Don't fail the pipeline if a staging deployment doesn't already exist.

inputs:

    continueOnError: true

inputs:

    azureSubscription: \$(azureSubscription)

    Action: 'Delete Staging Deployment'

    AppName: customer-api

    AzureSpringCloud: contoso-dev-az-spr-cld

## Deploying

### To production

The following example deploys to the default production deployment in Azure Spring Cloud. This is the only possible deployment scenario when using the Basic SKU:

variables:

```
azureSubscription: Contoso
```

steps:

```
- task: AzureSpringCloud@0
```

inputs:

```
azureSubscription: $(azureSubscription)
```

```
Action: 'Deploy'
```

```
AzureSpringCloud: contoso-dev-az-spr-cld
```

```
AppName: customer-api
```

```
UseStagingDeployment: false
```

```
DeploymentName: default
```

```
Package: '$(System.DefaultWorkingDirectory)/**/*customer-api*.jar'
```

## Blue-green

The following example deploys to a pre-existing staging deployment. This deployment will not receive production traffic until it is set as a production deployment.

variables:

```
azureSubscription: Contoso
```

steps:

```
- task: AzureSpringCloud@0
```

inputs:

```
azureSubscription: $(azureSubscription)
```

```
Action: 'Deploy'
```

```
AzureSpringCloud: contoso-dev-az-spr-cld  
AppName: customer-api  
UseStagingDeployment: true  
Package: '$(System.DefaultWorkingDirectory)/**/*customer-api*.jar'
```

## Setting production deployment

The following example will set the current staging deployment as production, effectively swapping which deployment will receive production traffic.

variables:

```
azureSubscription: Contoso
```

steps:

```
- task: AzureSpringCloud@0
```

inputs:

```
azureSubscription: $(azureSubscription)
```

```
Action: 'Set Production'
```

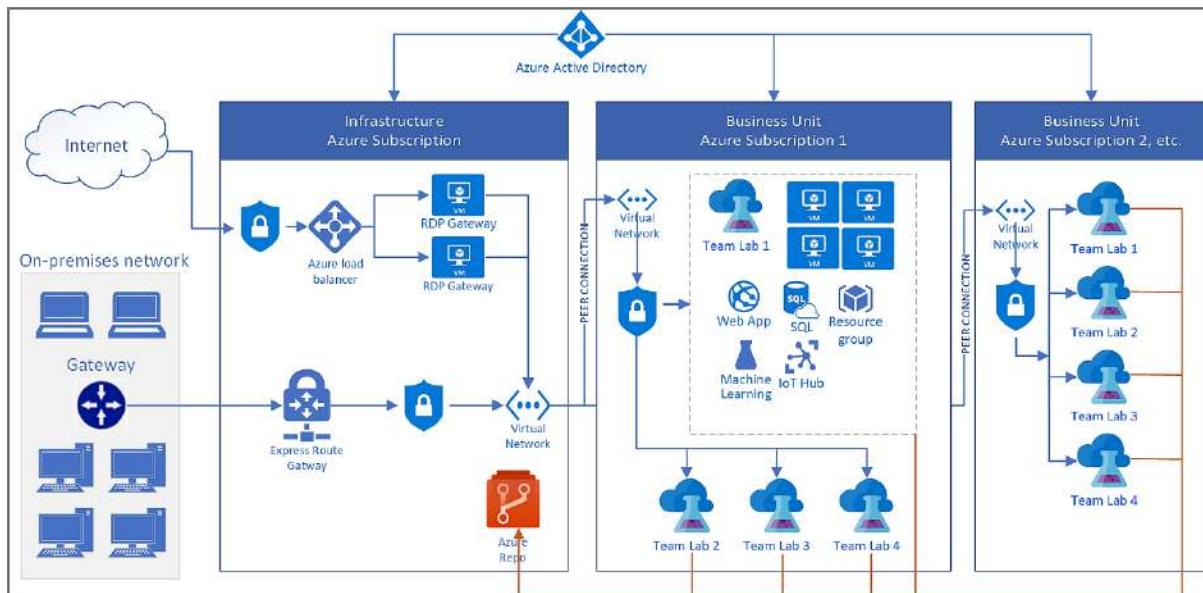
```
AzureSpringCloud: contoso-dev-az-spr-cld
```

```
AppName: customer-api
```

```
UseStagingDeployment: true
```

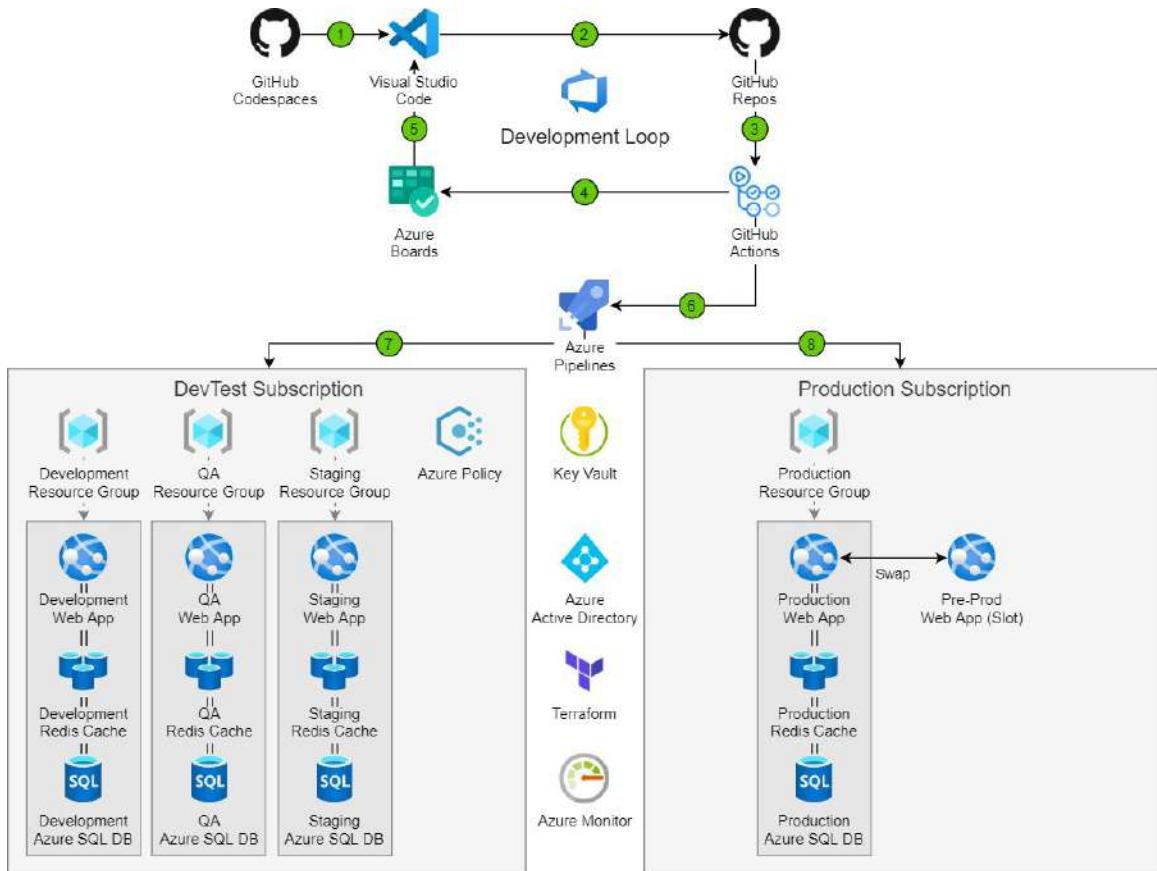
## AZURE DEVTEST LAB

Azure DevTest Labs is a service that enables developers to efficiently self-manage virtual machines (VMs) and Platform as a service (PaaS) resources without waiting for approvals. DevTest Labs creates labs consisting of pre-configured bases or Azure Resource Manager templates. These labs have all the necessary tools and software that you can use to create environments.



By using DevTest Labs, you can test the latest versions of your applications by doing the following tasks:

- Quickly create Windows and Linux environments by using reusable templates and artifacts.
- Easily integrate your deployment pipeline with DevTest Labs to create on-demand environments.
- Scale up your load testing by creating multiple test agents and pre-prepared environments for training and demos.



## Cost control and governance

DevTest Labs makes it easier to control costs by allowing you to do the following tasks:

- Set policies on your labs, such as number of VMs per user or per lab.
- Create policies to automatically shut down and start VMs.
- Track costs on VMs and PaaS resources spun up inside labs to stay within your budget. Receive notice of high-projected costs for labs so you can take necessary actions.
- Stay within the context of your labs so you don't spin up resources outside of them.

## Quickly get to ready-to-test

DevTest Labs lets you create pre-provisioned environments to develop and test applications. Just claim the environment of your application's last good build and start working. Or use containers for even faster, leaner environment creation.

## Create once, use everywhere

Capture and share PaaS environment templates and artifacts within your team or organization—all in source control—to easily create developer and test environments.

## Worry-free self-service

DevTest Labs enables your developers and testers to quickly and easily create IaaS VMs and PaaS resources by using a set of pre-configured resources.

## Use IaaS and PaaS resources

Spin up resources, such as Azure Service Fabric clusters, or SharePoint farms, by using Resource Manager templates. The templates come from the public environment repository or connect the lab to your own Git repository. You can also spin up an empty resource group (sandbox) by using a Resource Manager template to explore Azure within the context of a lab.

## Integrate with your existing toolchain

Use pre-made plug-ins or the API to create development/testing environments directly from your preferred continuous integration (CI) tool, integrated development environment (IDE), or automated release pipeline. You can also use the comprehensive command-line tool.

## Lab

A lab is the infrastructure that encompasses a group of resources, such as Virtual Machines (VMs), that lets you better manage those resources by specifying limits and quotas.

## Virtual machine

An Azure VM is one type of on-demand, scalable computing resources that Azure offers. Azure VMs give you the flexibility of virtualization without having to buy and maintain the physical hardware that runs it.

Overview of Windows virtual machines in Azure gives you information to consider before you create a VM, how you create it, and how you manage it.

## **Claimable VM**

An Azure Claimable VM is a virtual machine available to any lab user with permissions. Lab admins can prepare VMs with specific base images and artifacts and then save them to a shared pool. Lab users can claim a VM from the pool when they need one with that specific configuration.

A VM that is claimable isn't initially assigned to any particular user, but will show up in every user's list under "Claimable virtual machines". After a VM is claimed by a user, it's moved up to **My virtual machines** and is no longer claimable by any other user.

## **Environment**

In DevTest Labs, an environment refers to a collection of Azure resources in a lab. Create an environment discusses how to create multi-VM environments from your Azure Resource Manager templates.

## **Base images**

Base images are VM images with all the tools and settings preinstalled and configured. You can create a VM by picking an existing base and adding an artifact to install your test agent. The use of bases images reduces VM creation time.

## **Artifacts**

Artifacts are used to deploy and configure your application after a VM is provisioned. Artifacts can be:

- Tools that you want to install on the VM - such as agents, Fiddler, and Visual Studio.
- Actions that you want to run on the VM - such as cloning a repo.
- Applications that you want to test.

Artifacts are Azure Resource Manager JSON files that contain instructions to deploy and apply configurations.

## **Artifact repositories**

Artifact repositories are git repositories where artifacts are checked in. Artifact repositories can be added to multiple labs in your organization enabling reuse and sharing.

## Formulas

Formulas provide a mechanism for fast VM provisioning. A formula in DevTest Labs is a list of default property values used to create a lab VM. With formulas, VMs with the same set of properties - such as base image, VM size, virtual network, and artifacts - can be created without needing to specify those properties each time. When creating a VM from a formula, the default values can be used as-is or modified.

## Policies

Policies help in controlling cost in your lab. For example, you can create a policy to automatically shut down VMs based on a defined schedule.

## Caps

Caps is a mechanism to minimize waste in your lab. For example, you can set a cap to restrict the number of VMs that can be created per user, or in a lab.

## Security levels

Security access is determined by Azure role-based access control (Azure RBAC). To understand how access works, it helps to understand the differences between a permission, a role, and a scope as defined by Azure RBAC.

Term	Description
Permission	A defined access to a specific action (for example, read-access to all virtual machines).
Role	A set of permissions that can be grouped and assigned to a user. For example, the <i>subscription owner</i> role has access to all resources within a subscription.
Scope	A level within the hierarchy of an Azure resource, such as a resource group, a single lab, or the entire subscription.

Within the scope of DevTest Labs, there are two types of roles to define user permissions: lab owner and lab user.

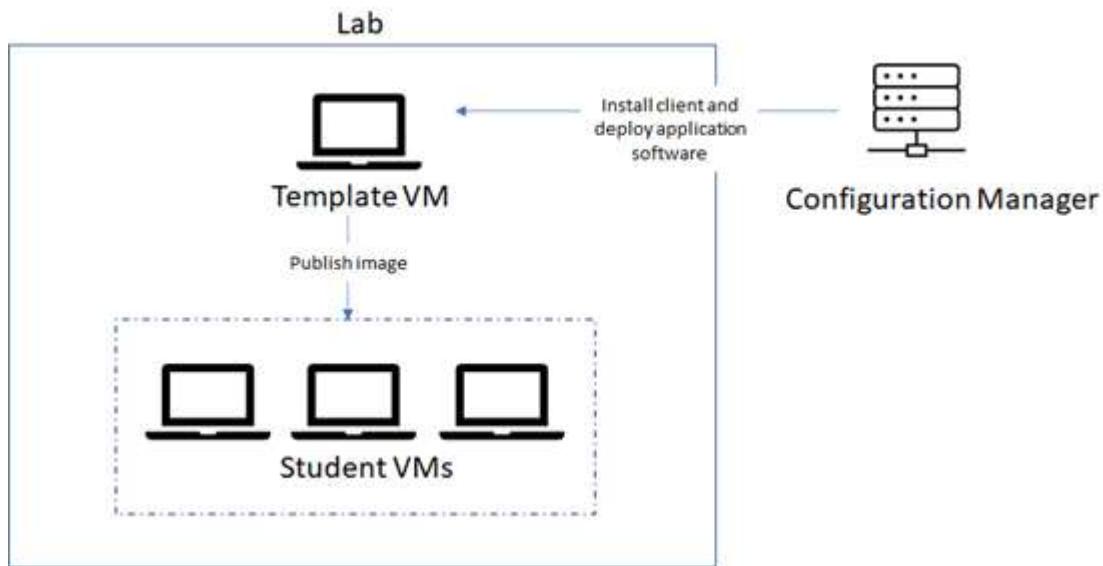
<b>Role</b>	<b>Description</b>
Lab Owner	Has access to any resources within the lab. A lab owner can modify policies, read and write any VMs, change the virtual network, and so on.
Lab User	Can view all lab resources, such as VMs, policies, and virtual networks, but can't modify policies or any VMs created by other users.

Since scopes are hierarchical, when a user has permissions at a certain scope, they also have permissions at every lower-level scope. Subscription owners have access to all resources in a subscription, which include virtual machines, virtual networks, and labs. A subscription owner automatically inherits the role of lab owner. However, the opposite isn't true. A lab owner has access to a lab, which is a lower scope than the subscription level. So, a lab owner can't see virtual machines or virtual networks or any resources that are outside of the lab.

## AZURE LAB SERVICES

Azure Lab Services enables you to quickly set up a classroom lab environment in the cloud. An educator creates a classroom lab, provisions Windows, or Linux virtual machines, installs the necessary software and tools labs in the class, and makes them available to students. The students in the class connect to virtual machines (VMs) in the lab, and use them for their projects, assignments, classroom exercises.

Currently, classroom lab is the only type of managed lab that's supported by Azure Lab Services. The service itself handles all the infrastructure management for a managed lab type, from spinning up VMs to handling errors and scaling the infrastructure. You specify what kind of infrastructure you need and install any tools or software that's required for the class. Learn more about service architecture.



After an IT admin creates a lab account in Azure Lab Services, an instructor can quickly set up a lab for the class, specify the number and type of VMs that are needed for exercises in the class, and add users to the class. Once a user registers to the class, the user can access the VM to do exercises for the class.

# Azure Lab Services - General flow

## Step 1: IT Management



- Create Lab Accounts
- Define custom images for the Labs
- Make images available for the Labs
- Assign professors/tutors to their Labs
- Integrates Labs to existing VNets

## Step 2: Professors/Tutors Management



- Create new Labs as needed
- Manage Lab usage: scheduling, quotas
- Creates VM templates for each Lab
- Manages students invitation

## Step 3: Students



- Get registered and joined into a Lab
- Can see all Labs available
- Connect to Lab's VMs via RDP

## Key capabilities

Azure Lab Services supports the following key capabilities/features:

- **Fast and flexible setup of a lab.** Using Azure Lab Services, lab owners can quickly set up a lab for their needs. The service offers the option to take care of all Azure infrastructure work for managed lab types. The service provides built-in scaling and resiliency of infrastructure for labs that the service manages for you.
- **Simplified experience for lab users.** Users who are invited to your lab get immediate access to the resources you give them inside your labs. They just need to sign in to see the full list of virtual machines they have access to across multiple labs. They can click on a single button to connect to the virtual machines and start working. Users don't need Azure subscriptions to use the service. Lab users can register to a lab with a registration code and can access the lab anytime to use the lab's resources.
- **Cost optimization and analysis.** Keep your budget in check by controlling exactly how many hours your lab users can use the virtual machines. Set up schedules in the lab to allow users to use the virtual machines only during designated time slots or set up reoccurring auto-shutdown and start times. Keep track of individual users' usage and set limits.
- **Automatic management of Azure infrastructure and scale** Azure Lab Services is a managed service, which means that provisioning and management of a lab's underlying infrastructure is handled automatically by the service. You can just focus on preparing the right lab experience for your users. Let the service handle the rest

and roll out your lab's virtual machines to your audience. Scale your lab to hundreds of virtual machines with a single click.

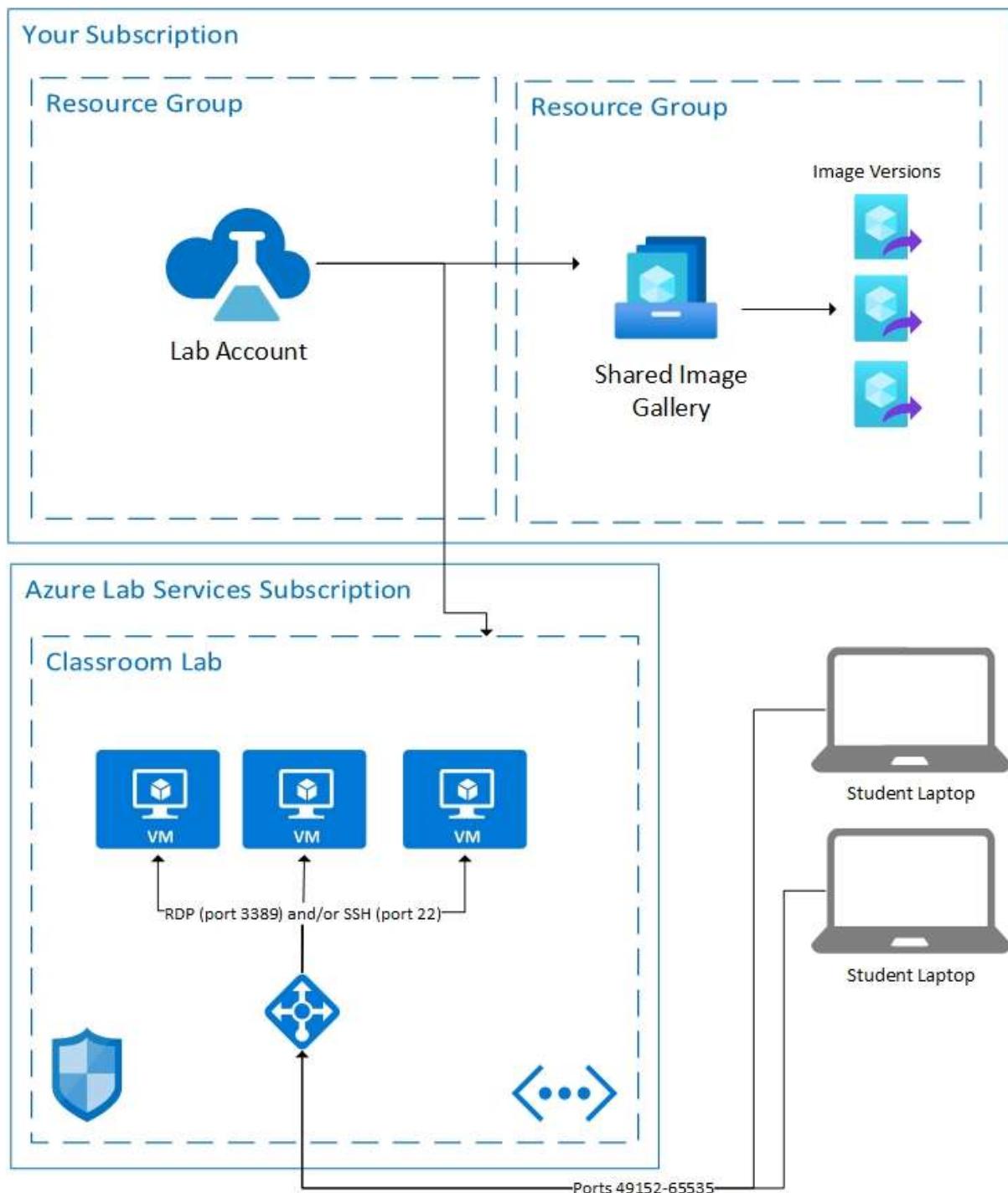
If you want to just input what you need in a lab and let the service set up and manage infrastructure required for the lab, choose from one of the **managed lab types**. Currently, **classroom lab** is the only managed lab type that you can create with Azure Lab Services.

The following sections provide more details about these labs.

## Managed lab types

Azure Lab Services allows you to create labs whose infrastructure is managed by Azure. This article refers to them as managed lab types. Managed lab types offer different types of labs that fit for your specific need. Currently, the only managed lab type that's supported is **classroom lab**.

Managed lab types enable you to get started right away, with minimal setup. The service itself handles all the management of the infrastructure for the lab, from spinning up the VMs to handling errors and scaling the infrastructure. To create a managed lab type such as a classroom lab, you need to create a lab account for your organization first. The lab account serves as the central account in which all labs in the organization are managed.



When you create and use Azure resources in these managed lab types, the service creates and manages resources in internal Microsoft subscriptions. They are not created in your own Azure subscription. The service keeps track of usage of these resources in internal Microsoft subscriptions. This usage is billed back to your Azure subscription that contains the lab account.

Here are some of the **use cases for managed lab types**:

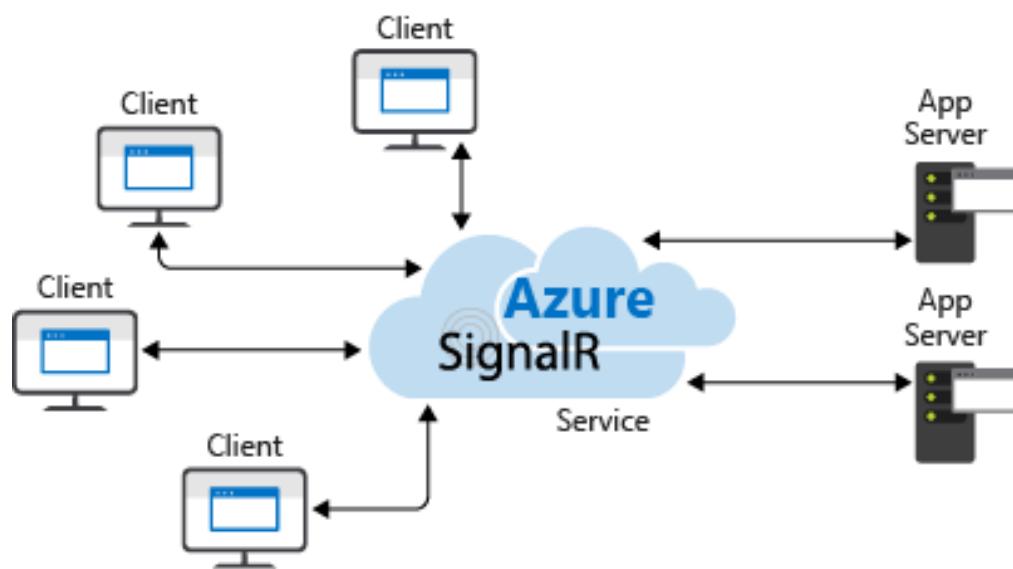
- Provide students with a lab of virtual machines that are configured with exactly what's needed for a class. Give each student a limited number of hours for using the VMs for homework or personal projects.
- Set up a pool of high performance compute VMs to perform compute-intensive or graphics-intensive research. Run the VMs as needed, and clean up the machines once you are done.
- Move your school's physical computer lab into the cloud. Automatically scale the number of VMs only to the maximum usage and cost threshold that you set on the lab.
- Quickly provision a lab of virtual machines for hosting a hackathon. Delete the lab with a single click once you're done.

## Example class types

You can set up labs for several types of classes with Azure Lab Services. See the Example class types on Azure Lab Services article for a few example types of classes for which you can set up labs with Azure Lab Services.

## SIGNALR SERVICES

Azure SignalR Service simplifies the process of adding real-time web functionality to applications over HTTP. This real-time functionality allows the service to push content updates to connected clients, such as a single page web or mobile application. As a result, clients are updated without the need to poll the server, or submit new HTTP requests for updates.



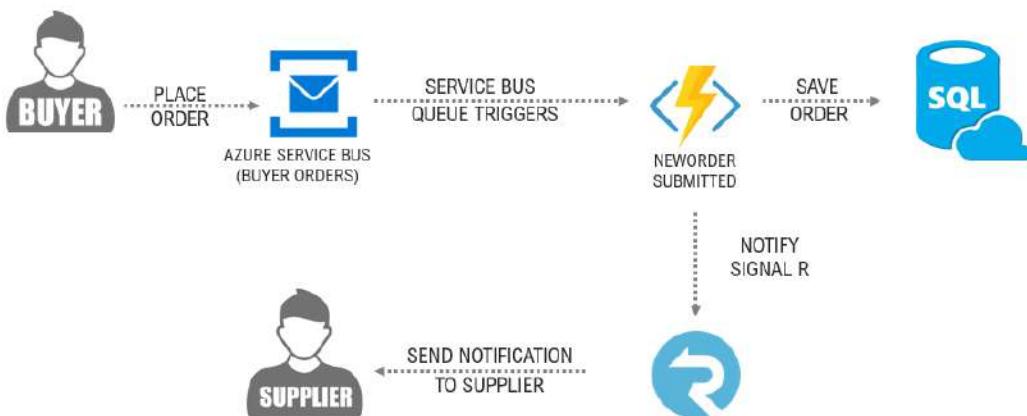
## What is Azure SignalR Service used for?

Any scenario that requires pushing data from server to client in real time, can use Azure SignalR Service.

Traditional real-time features that often require polling from server, can also use Azure SignalR Service.

Azure SignalR Service has been used in a wide variety of industries, for any application type that requires real-time content updates. We list some examples that are good to use Azure SignalR Service:

- **High frequency data updates:** gaming, voting, polling, auction.
- **Dashboards and monitoring:** company dashboard, financial market data, instant sales update, multi-player game leader board, and IoT monitoring.
- **Chat:** live chat room, chat bot, on-line customer support, real-time shopping assistant, messenger, in-game chat, and so on.
- **Real-time location on map:** logistic tracking, delivery status tracking, transportation status updates, GPS apps.
- **Real time targeted ads:** personalized real time push ads and offers, interactive ads.
- **Collaborative apps:** coauthoring, whiteboard apps and team meeting software.
- **Push notifications:** social network, email, game, travel alert.
- **Real-time broadcasting:** live audio/video broadcasting, live captioning, translating, events/news broadcasting.
- **IoT and connected devices:** real-time IoT metrics, remote control, real-time status, and location tracking.
- **Automation:** real-time trigger from upstream events.



## **What are the benefits using Azure SignalR Service?**

### **Standard based:**

SignalR provides an abstraction over a number of techniques used for building real-time web applications. WebSockets is the optimal transport, but other techniques like Server-Sent Events (SSE) and Long Polling are used when other options aren't available. SignalR automatically detects and initializes the appropriate transport based on the features supported on the server and client.

### **Native ASP.NET Core support:**

SignalR Service provides native programming experience with ASP.NET Core and ASP.NET. Developing new SignalR application with SignalR Service, or migrating from existing SignalR based application to SignalR Service requires minimal efforts. SignalR Service also supports ASP.NET Core's new feature, Server-side Blazor.

### **Broad client support:**

SignalR Service works with a broad range of clients, such as web and mobile browsers, desktop apps, mobile apps, server process, IoT devices, and game consoles. SignalR Service offers SDKs in different languages. In addition to native ASP.NET Core or ASP.NET C# SDKs, SignalR Service also provides JavaScript client SDK, to enable web clients, and many JavaScript frameworks. Java client SDK is also supported for Java applications, including Android native apps. SignalR Service supports REST API, and serverless through integrations with Azure Functions and Event Grid.

### **Handle large-scale client connections:**

SignalR Service is designed for large-scale real-time applications. SignalR Service allows multiple instances to work together to scale to millions of client connections. The service also supports multiple global regions for sharding, high availability, or disaster recovery purposes.

### **Remove the burden to self-host SignalR:**

Compared to self-hosted SignalR applications, switching to SignalR Service will remove the need to manage back planes that handle the scales and client connections. The fully managed service also simplifies web applications and saves hosting cost. SignalR

Service offers global reach and world-class data center and network, scales to millions of connections, guarantees SLA, while providing all the compliance and security at Azure standard.

## **Offer rich APIs for different messaging patterns:**

SignalR Service allows the server to send messages to a particular connection, all connections, or a subset of connections that belong to a specific user, or have been placed in an arbitrary group.

## **How to use Azure SignalR Service**

There are many different ways to program with Azure SignalR Service, as some of the samples listed here:

- **Scale an ASP.NET Core SignalR App** - Integrate Azure SignalR Service with an ASP.NET Core SignalR application to scale out to hundreds of thousands of connections.
- **Build serverless real-time apps** - Use Azure Functions' integration with Azure SignalR Service to build serverless real-time applications in languages such as JavaScript, C#, and Java.
- **Send messages from server to clients via REST API** - Azure SignalR Service provides REST API to enable applications to post messages to clients connected with SignalR Service, in any REST capable programming languages.

## **Create a chat room by using SignalR Service**

Azure SignalR Service is an Azure service that helps developers easily build web applications with real-time features. This service was originally based on SignalR for ASP.NET Core 2.1, but now supports later versions.

This app will make a connection with your Azure SignalR Service resource to enable real-time content updates. You'll host the web application locally and connect with multiple browser clients. Each client will be able to push content updates to all other clients.

You can use any code editor to complete the steps in this quickstart. One option is Visual Studio Code, which is available on the Windows, macOS, and Linux platforms.

### Create an Azure SignalR resource

1. To create an Azure SignalR Service resource, first sign in to the Azure portal. In the upper-left side of the page, select **+ Create a resource**. In the **Search the Marketplace** text box, enter **SignalR Service**.
2. Select **SignalR Service** in the results, and select **Create**.
3. On the new **SignalR** settings page, add the following settings for your new SignalR resource:

Name	Recommended value	Description
Resource name	<i>testsignalr</i>	Enter a unique resource name to use for the SignalR resource. The name must be a string of 1 to 63 characters and contain only numbers, letters, and the hyphen (-) character. The name cannot start or end with the hyphen character, and consecutive hyphen characters are not valid.
Subscription	Choose your subscription	Select the Azure subscription that you want to use to test SignalR. If your account has only one subscription, it's automatically selected and the <b>Subscription</b> drop-down isn't displayed.
Resource group	Create a resource group named <i>SignalRTTestResources</i>	Select or create a resource group for your SignalR resource. This group is useful for organizing multiple resources that you might want to delete at the same time by deleting the resource group. For more information, see <a href="#">Using resource groups to manage your Azure resources</a> .
Location	<i>East US</i>	Use <b>Location</b> to specify the geographic location in which your SignalR resource is hosted. For the best performance, we recommend that you create the resource in the same region as other components of your application.

Name	Recommended value	Description
Pricing tier	Free	Currently, <b>Free</b> and <b>Standard</b> options are available.
Pin to dashboard	<input checked="" type="checkbox"/>	Select this box to have the resource pinned to your dashboard so it's easier to find.

4. Select Review + create. Wait for the validation to complete.
5. Select Create. The deployment might take a few minutes to complete.
6. After the deployment is complete, select Keys under SETTINGS. Copy your connection string for the primary key. You'll use this string later to configure your app to use the Azure SignalR Service resource.

The connection string will have the following form:

```
Endpoint=<service_endpoint>;AccessKey=<access_key>;
```

## Create an ASP.NET Core web app

You can use the .NET Core command-line interface (CLI) to create an ASP.NET Core MVC web app project. The advantage of using the .NET Core CLI over Visual Studio is that it's available across the Windows, macOS, and Linux platforms.

1. Create a folder for your project. This quickstart uses the *E:\Testing\chattest* folder.
2. In the new folder, run the following command to create the project:

```
dotnet new mvc
```

## Add Secret Manager to the project

In this section, you'll add the Secret Manager tool to your project. The Secret Manager tool stores sensitive data for development work outside your project tree. This approach helps prevent the accidental sharing of app secrets in source code.

1. Open your *.csproj* file. Add a `DotNetCliToolReference` element to include `Microsoft.Extensions.SecretManager.Tools`. Also add a `UserSecretsId` element as shown in the following code for *chattest.csproj*, and save the file.

```

<Project Sdk="Microsoft.NET.Sdk.Web">

    <PropertyGroup>
        <TargetFramework>netcoreapp3.1</TargetFramework>
        <UserSecretsId>SignalRChatRoomEx</UserSecretsId>
    </PropertyGroup>

    <ItemGroup>
        <DotNetCliToolReference Include="Microsoft.VisualStudio.Web.CodeGeneration.Tools" Version="2.0.4" />
        <DotNetCliToolReference Include="Microsoft.Extensions.SecretManager.Tools" Version="2.0.2" />
    </ItemGroup>

</Project>

```

## Add Azure SignalR to the web app

1. Add a reference to the Microsoft.Azure.SignalR NuGet package by running the following command:

```
dotnet add package Microsoft.Azure.SignalR
```

2. Run the following command to restore packages for your project:

```
dotnet restore
```

3. Add a secret named Azure:SignalR:ConnectionString to Secret Manager.

This secret will contain the connection string to access your SignalR Service resource. Azure:SignalR:ConnectionString is the default configuration key that SignalR looks for to establish a connection. Replace the value in the following command with the connection string for your SignalR Service resource. You must run this command in the same directory as the .csproj file.

```
dotnet user-secrets set Azure:SignalR:ConnectionString "<Your connection string>"
```

Secret Manager will be used only for testing the web app while it's hosted locally. In a later tutorial, you'll deploy the chat web app to Azure. After the web app is deployed to Azure, you'll use an application setting instead of storing the connection string with Secret Manager.

This secret is accessed with the Configuration API. A colon (:) works in the configuration name with the Configuration API on all supported platforms.

4. Open *Startup.cs* and update the `ConfigureServices` method to use Azure SignalR Service by calling the `AddSignalR()` and `AddAzureSignalR()` methods:

```
public void ConfigureServices(IServiceCollection services)
{
    services.AddSignalR()
        .AddAzureSignalR();
}
```

By not passing a parameter to `AddAzureSignalR()`, this code uses the default configuration key for the SignalR Service resource connection string. The default configuration key is *Azure:SignalR:ConnectionString*.

5. In *Startup.cs*, update the `Configure` method by replacing it with the following code.

```
public void Configure(IApplicationBuilder app, IWebHostEnvironment env)
{
    app.UseRouting();

    app.UseFileServer();

    app.UseEndpoints(endpoints =>
    {
        endpoints.MapHub<ChatHub>("/chat");

    });
}
```

## Add a hub class

In SignalR, a hub is a core component that exposes a set of methods that can be called from the client. In this section, you define a hub class with two methods:

- Broadcast: This method broadcasts a message to all clients.
- Echo: This method sends a message back to the caller.

Both methods use the `Clients` interface that the ASP.NET Core SignalR SDK provides. This interface gives you access to all connected clients, so you can push content to your clients.

1. In your project directory, add a new folder named `Hub`. Add a new hub code file named `ChatHub.cs` to the new folder.
2. Add the following code to `ChatHub.cs` to define your hub class and save the file.

Update the namespace for this class if you used a project name that differs from `SignalR.Mvc`.

```
using Microsoft.AspNetCore.SignalR;
using System.Threading.Tasks;

namespace SignalR.Mvc
{
    public class ChatHub : Hub
    {
        public Task BroadcastMessage(string name, string message) =>
            Clients.All.SendAsync("broadcastMessage", name, message);

        public Task Echo(string name, string message) =>
            Clients.Client(Context.ConnectionId)
                .SendAsync("echo", name, $"{message} (echo from server)");
    }
}
```

## Add the client interface for the web app

The client user interface for this chat room app will consist of HTML and JavaScript in a file named *index.html* in the *wwwroot* directory.

Copy the *css/site.css* file from the *wwwroot* folder of the samples repository. Replace your project's *css/site.css* with the one you copied.

Here's the main code of *index.html*:

Create a new file in the *wwwroot* directory named *index.html*, copy, and paste the following HTML into the newly created file:

```
<!DOCTYPE html>

<html>
  <head>
    <link href="https://cdn.jsdelivr.net/npm/bootstrap@3.3.7/dist/css/bootstrap.min.css" rel="stylesheet" />
    <link href="css/site.css" rel="stylesheet" />
    <title>Azure SignalR Group Chat</title>
  </head>
  <body>
    <h2 class="text-center" style="margin-top: 0; padding-top: 30px; padding-bottom: 30px;">Azure SignalR Group Chat</h2>
    <div class="container" style="height: calc(100% - 110px);>
      <div id="messages" style="background-color: whitesmoke; "></div>
      <div style="width: 100%; border-left-style: ridge; border-right-style: ridge;">
        <textarea id="message"
          style="width: 100%; padding: 5px 10px; border-style: hidden;">
          placeholder="Type message and press Enter to send..."</textarea>
      </div>
      <div style="overflow: auto; border-style: ridge; border-top-style: hidden;">
        <button class="btn-warning pull-right" id="echo">Echo</button>
        <button class="btn-success pull-right" id="sendmessage">Send</button>
      </div>
    </div>
  </body>

```

```
</div>

<div class="modal alert alert-danger fade" id="myModal" tabindex="-1" role="dialog"
aria-labelledby="myModalLabel">

<div class="modal-dialog" role="document">

<div class="modal-content">

<div class="modal-header">

<div>Connection Error...</div>

<div><strong style="font-size: 1.5em;">Hit Refresh/F5</strong> to rejoin. ;)</
div>

</div>

</div>

</div>
```

```
<!--Reference the SignalR library.-->
```

```
<script src="https://cdn.jsdelivr.net/npm/@microsoft/signalr@3.1.8/dist/browser/
signalr.min.js"></script>
```

```
<!--Add script to update the page and send messages.-->
```

```
<script type="text/javascript">
document.addEventListener('DOMContentLoaded', function () {
```

```
const generateRandomName = () =>
Math.random().toString(36).substring(2, 10);
```

```
let username = generateRandomName();
const promptMessage = 'Enter your name:';
do {
    username = prompt(promptMessage, username);
```

```

        if (!username || username.startsWith('_') || username.indexOf('<') > -1 || username.indexOf('>') > -1) {

            username = "";

            promptMessage = 'Invalid input. Enter your name:';

        }

    } while (!username)

const messageInput = document.getElementById('message');

messageInput.focus();

function createMessageEntry(encodedName, encodedMsg) {

    var entry = document.createElement('div');

    entry.classList.add("message-entry");

    if (encodedName === "_SYSTEM_") {

        entry.innerHTML = encodedMsg;

        entry.classList.add("text-center");

        entry.classList.add("system-message");

    } else if (encodedName === "_BROADCAST_") {

        entry.classList.add("text-center");

        entry.innerHTML = `<div class="text-center broadcast-message">$
{encodedMsg}</div>`;

    } else if (encodedName === username) {

        entry.innerHTML = `<div class="message-avatar pull-right">${encodedName}
</div>` +

            `<div class="message-content pull-right">${encodedMsg}</div>`;

    } else {

        entry.innerHTML = `<div class="message-avatar pull-left">${encodedName}<
/div>` +

            `<div class="message-content pull-left">${encodedMsg}</div>`;
    }
}

```

```

    }

    return entry;
}

function bindConnectionMessage(connection) {
    var messageCallback = function (name, message) {
        if (!message) return;

        var encodedName = name;
        var encodedMsg = message.replace(/&/g, "&").replace(/</g,
"&lt;").replace(/>/g, "&gt;");

        var messageEntry = createMessageEntry(encodedName, encodedMsg);

        var messageBox = document.getElementById('messages');
        messageBox.appendChild(messageEntry);
        messageBox.scrollTop = messageBox.scrollHeight;
    };

    connection.on('broadcastMessage', messageCallback);
    connection.on('echo', messageCallback);
    connection.onclose(onConnectionError);
}

function onConnected(connection) {
    console.log('connection started');

    connection.send('broadcastMessage', '_SYSTEM_', username + ' JOINED');

    document.getElementById('sendmessage').addEventListener('click', function
(event) {
        if (messageInput.value) {

            connection.send('broadcastMessage', username, messageInput.value);
        }
    });
}

```

```
    messageInput.value = "";
    messageInput.focus();
    event.preventDefault();
});

document.getElementById('message').addEventListener('keypress', function (event)
{
    if (event.keyCode === 13) {
        event.preventDefault();
        document.getElementById('sendmessage').click();
        return false;
    }
});

document.getElementById('echo').addEventListener('click', function (event) {
    connection.send('echo', username, messageInput.value);

    messageInput.value = "";
    messageInput.focus();
    event.preventDefault();
});

function onConnectionError(error) {
    if (error && error.message) {
        console.error(error.message);
    }
}

var modal = document.getElementById('myModal');
modal.classList.add('in');
```

```

        modal.style = 'display: block;';

    }

const connection = new signalR.HubConnectionBuilder()

    .withUrl('/chat')

    .build();

bindConnectionMessage(connection);

connection.start()

    .then(() => onConnected(connection))

    .catch(error => console.error(error.message));

});

</script>

</body>

</html>

```

The code in *index.html* calls `HubConnectionBuilder.build()` to make an HTTP connection to the Azure SignalR resource.

If the connection is successful, that connection is passed to `bindConnectionMessage`, which adds event handlers for incoming content pushes to the client.

`HubConnection.start()` starts communication with the hub. Then, `onConnected()` adds the button event handlers. These handlers use the connection to allow this client to push content updates to all connected clients.

## Add a development runtime profile

In this section, you'll add a development runtime environment for ASP.NET Core.

1. Create a folder named *Properties* in your project.

```
{  
  "profiles": {  
    "ChatRoom": {  
      "commandName": "Project",  
      "launchBrowser": true,  
      "environmentVariables": {  
        "ASPNETCORE_ENVIRONMENT": "Development"  
      },  
      "applicationUrl": "http://localhost:5000/"  
    },  
  }  
}
```

## Build and run the app locally

1. To build the app by using the .NET Core CLI, run the following command in the command shell:

```
dotnet build
```

2. After the build successfully finishes, run the following command to run the web app locally:

```
dotnet run
```

The app will be hosted locally on port 5000, as configured in our development runtime profile:

```
info: Microsoft.Hosting.Lifetime[0]  
      Now listening on: https://localhost:5001  
info: Microsoft.Hosting.Lifetime[0]  
      Now listening on: http://localhost:5000  
info: Microsoft.Hosting.Lifetime[0]  
      Application started. Press Ctrl+C to shut down.  
info: Microsoft.Hosting.Lifetime[0]  
      Hosting environment: Development  
info: Microsoft.Hosting.Lifetime[0]  
      Content root path: E:\Testing\chattest
```

3. Open two browser windows. In each browser, go to <http://localhost:5000>. You're prompted to enter your name. Enter a client name for both clients and test pushing message content between both clients by using the Send button.

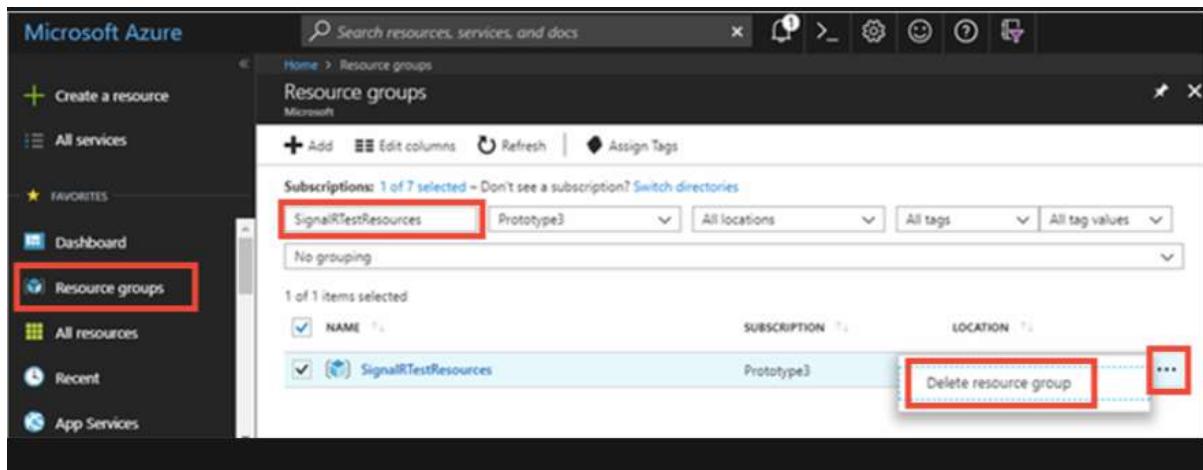
## Clean up resources

If you'll continue to the next tutorial, you can keep the resources created in this quickstart and reuse them.

If you're finished with the quickstart sample application, you can delete the Azure resources created in this quickstart to avoid charges.

Sign in to the Azure portal and select **Resource groups**.

In the **Filter by name** text box, type the name of your resource group. The instructions for this quickstart used a resource group named *SignalRTTestResources*. On your resource group in the result list, select the ellipsis (...) > **Delete resource group**.

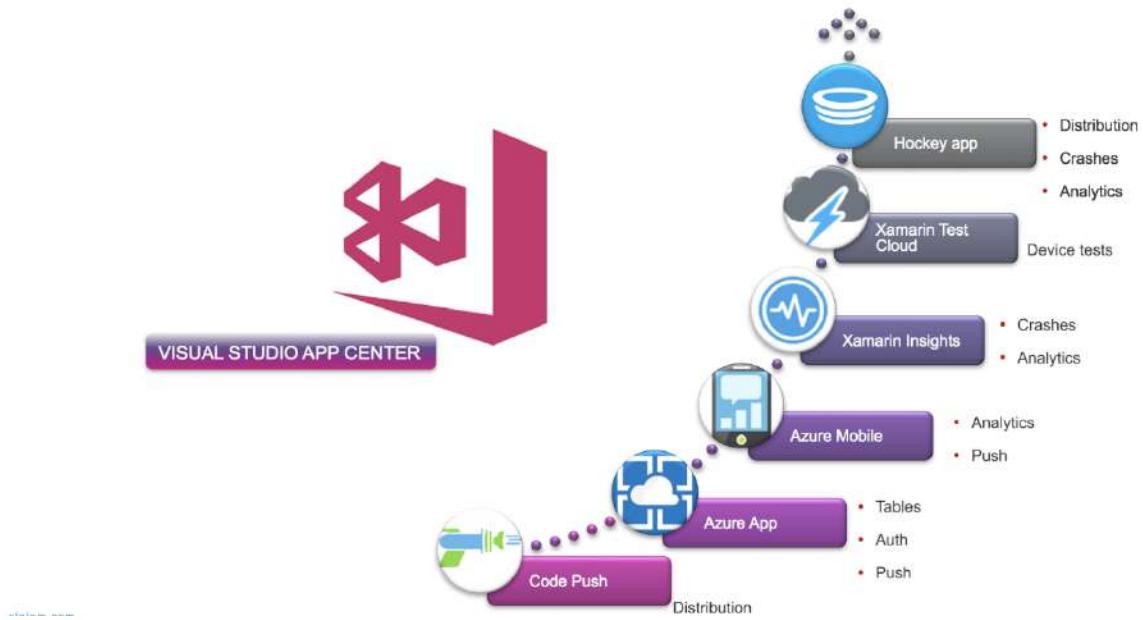


You're asked to confirm the deletion of the resource group. Enter the name of your resource group to confirm, and select **Delete**.

After a few moments, the resource group and all of its resources are deleted.

## VISUAL STUDIO APP CENTER

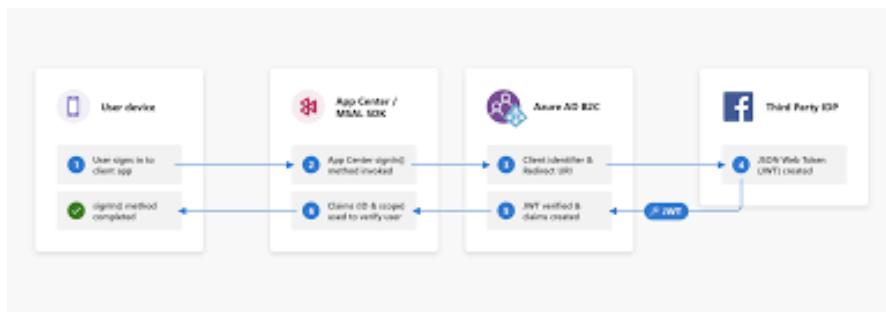
Visual Studio App Center brings together multiple common services into a DevOps cloud solution. Developers use App Center to Build, Test, and Distribute applications. Once the app's deployed, developers monitor the status and usage of the app using the Analytics and Diagnostics services.



## Set up your App Center account

### Apps

To get started, add an app to your App Center account or accept an invite to someone else's app. You can have different permission levels for each app: **Developer**, **Viewer**, or **Manager**.



### Creating an app

To create an app:

1. Log in to Visual Studio App Center.
2. Click the **Add new** dropdown in the upper-right corner of the page, then choose **Add new app**.

3. Populate the panel that appears with information about the new app.

## Release Type

Select one of the suggested release types: Alpha, Beta, Enterprise, Production, or Store. You can also use a custom release type by selecting the 'Custom' field. This custom release type must be a single word, alphanumeric, starting with a capital letter or number, followed by lowercase or numbers.

## Uploading an app icon

Upload an app icon in the **Add new app** dialog, or in the settings page of your app. Uploading an icon in App Center doesn't change the icon in the app bundle, meaning the icon of the app when viewed on the install portal and installed on devices won't reflect this change.

## Accessing apps

All apps that belong to you can be found in **My Apps**. When looking for apps owned by organizations you belong to, click on the organization in the left navigation.

## App secrets

App secret is like an API key for your app, it allows events and telemetry to be sent to App Center backend. It doesn't provide any access to your account. It can't be used to invoke App Center REST APIs (like trigger builds). If your code is open source, we recommend you inject the secret at build or in a similar way.

## App roles

On each app there are three roles:

- **Managers** can manage app settings, collaborators, and integrations.
- **Developers** can manage app services (e.g. create builds, run tests).
- **Viewers** can view and download all data but can't make changes.

For every app you create, whether owned by you or your organization, you're automatically assigned as Manager for the app. Additionally, all organization Admins are

assigned as managers for all apps within the organization. Collaborators and Members can be assigned any of the three roles listed above for each app. Learn more about managing organization roles.

## Changing app roles

To change the permission of a collaborator:

1. Select an app from the dashboard.
2. In the left side navigation, select **Settings**.
3. Select **Collaborators**.
4. Next to the name of the user you wish to change roles of, reassign the role by making a selection from the drop-down.

## Adding collaborators to apps

To share your app with others, select an app from the dashboard and then click **Manage app** to add collaborators by typing in the user's email address.

## Transferring an app

When you transfer an app into an organization, all app data will be transferred over. The admins of the new organization will also gain access to the app that was transferred in.

To transfer an app from your personal account to an organization or from one organization to another:

1. Select an app from the dashboard.
2. In the left side navigation, select **Settings**.
3. Select the 'More' menu, the three dots in the upper-right corner.
4. Select **Transfer app to organization**.
5. Select the organization you wish to transfer the app into.

## Deleting an app

You can delete any App Center app that you no longer need.

Follow these steps to delete any of your apps:

1. Select the app you want to delete from the dashboard.
2. In the left side navigation, select **Settings**.
3. Select the 'More' menu, the three dots in the upper-right corner.
4. Select **Delete app**.
5. Carefully consider the action, then choose **Delete app** or **Cancel**.

## Organizations

Create an Organization to invite **Collaborators** and **Administrators** to work on your apps, and manage their permissions.

### Organization roles

These are the three roles within an organization and the actions each can take:

Roles	Create App	See all apps	Manage settings	Add people	Manage Shared Distribution Groups and Teams
Admins	Yes	Yes	Yes	Yes	Yes
Collaborators	Yes	Only if you belong	No	No	Yes
Members	Yes	Only if you belong	No	No	No

- **Admins** are Managers of all the apps in the organization. Learn more about app permissions.
- When someone gets added to an app owned by an organization, they get invited as **Members** of the organization.

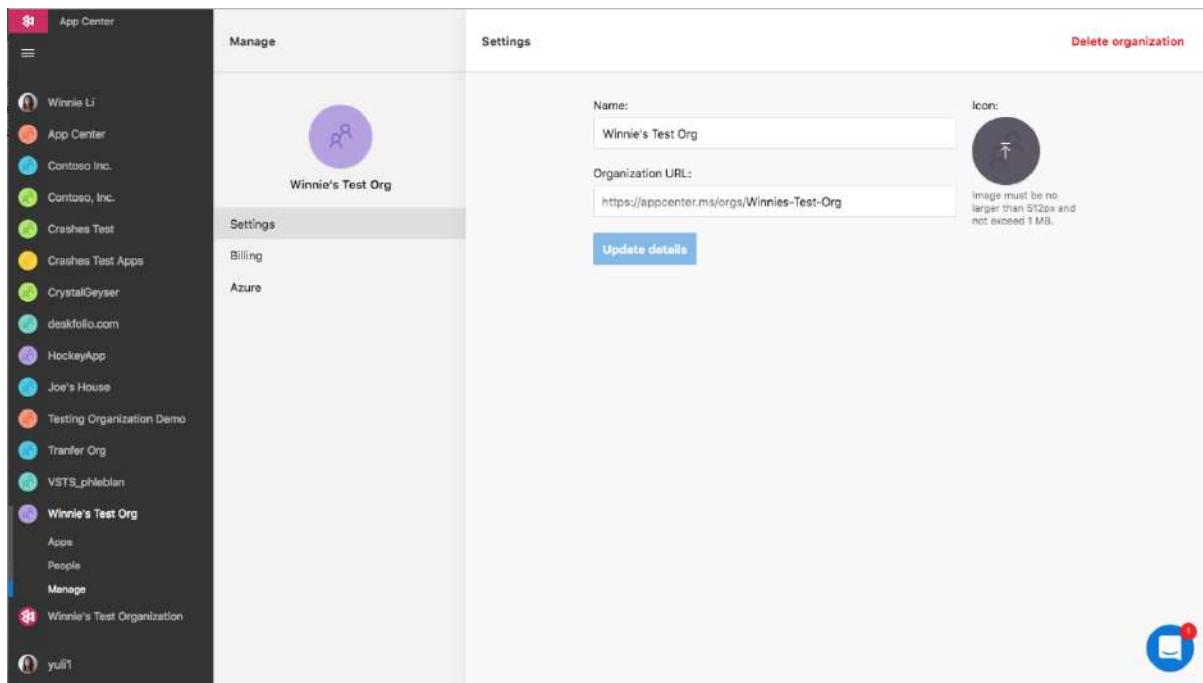
## Creating an organization

To create an organization, log in and click the **Add new** dropdown and choose **Add new organization**.

## Uploading an organization icon

Customize your organization by uploading an org icon:

1. On the left side navigation, select the organization
2. Select the **Manage** tab
3. On the right side of the screen, click the placeholder org icon
4. Select an image file no larger than 512px and doesn't exceed 1 MB



## Accessing organizations

All of your organizations are accessible in the left navigation.

## Adding users to an organization

There are two ways to add users to an organization:

### Directly to the organization:

1. On the left side navigation, select the organization
2. Select **People**
3. Type the user's email address to add the user

This gives the user access to the organization as a collaborator. However, they won't see any of the apps listed in the organization unless they're explicitly invited to the app or if they're an admin of the organization.

#### **In-directly through an app:**

1. Select an app within an organization
2. On the left side navigation, select the **Settings** page
3. Select **People**
4. Type the user's email address to add the user

Adding a user from outside the organization to an app automatically adds them to the organization. However, the user will only see the app(s) they were invited to.

#### **Changing users' organization roles**

1. On the left side navigation, select the organization
2. Select **People**
3. Select the user
4. Use the dropdown to re-assign the role

Only 'Admins' can change the role of collaborators and other admins.

#### **Removing users from an organization**

1. On the left side navigation, select the organization
2. Select **People**
3. Select the collaborator
4. Click **Remove from organization**

Removing a collaborator from an organization will remove the user from all apps within the organization.

## **Leaving an organization**

1. Click on the App Center user menu in the upper-right corner of any page
2. Select **Account Settings**
3. Select **Organizations**
4. Click the **Leave** button by the organization you want to leave.

## **Deleting an organization**

1. On the left side navigation, select the organization
2. Select **Manage**
3. On the upper right hand side, click the **Delete Organization** button

## **Teams**

Create Teams within your organization to better manage large organizations and permission settings.

### **Creating teams**

Any collaborator can create teams within an organization. Organizations can have multiple teams but each team belongs to only one organization. To create a team:

1. On the left side navigation, select your organization
2. Select **People**
3. Select **Teams**
4. Click on the **Add New Team** button
5. Enter a team name
6. Click **Create Team** button

### **Adding users to a team**

The team creator or any organization admin can invite other collaborators to a team. Users must join the organization before they can join a team.

### **Removing users from a team**

The team creator or any organization admin can remove users from a team.

## **Adding the team to an app**

The team creator, any app manager, or any organization admin can add a team to an app. When a team is added to an app, all users of the team gain access to the app with the team's assigned role.

There are two ways to add a team to an app:

### **In team setting**

1. On the left side navigation, select the team organization
2. Select **People**
3. Select **Teams**
4. Select the team you wish to add to an app
5. In the top navigation, select **Apps**
6. In the search box, type the app name you wish to add

### **In app setting**

1. In the dashboard, select the app you wish to add a team to
2. In the left navigation, select **Settings**
3. Click on **Collaborators**
4. Type the team you wish to add to the app

## **Removing the team from an app:**

The team creator or any organization admin can remove a team from an app and the team member's permission to the app will be revoked. There are two ways to remove a team from an app:

### **In team setting**

1. In the left side navigation, select the team organization
2. Select **People**.
3. Click on **Teams**
4. Select the team you wish to add to an app
5. Select **APPS** in the top navigation
6. Hover your cursor to the app you want to remove and click on the delete icon on the right

## In app setting

1. In the dashboard, select the app you wish to add a team to
2. In the left navigation, select **Settings**
3. Select **Collaborators**.
4. Hover your cursor to the app you want to remove and click on the delete icon on the right

## Changing Permissions

The team creator or any org admin can change the app permission of the team. Only admins can change permissions of an individual user.

### Permission levels

- **Manager** - invite members and access settings
- **Developer** - manage services (e.g. create builds, run tests)
- **Viewer** - view and download app data

#### Note

Users get the highest permission assigned (team level or app level).

## Leaving a team

Users can leave a team upon confirmation and will lose access to any apps associate with the team. The team creator can't leave the team. The only option is to delete the team.

## Deleting a team

The team creator or any admin can delete a team. All users in the team will have app permissions revoked.

## Manage your account preferences and settings

### Account Profile Photos

App Center uses Gravatar to manage user profile photos. To get started, create a Gravatar account using the same email for your App Center account. Once your account has been created, follow the instructions on Gravatar's website to pick your new profile image. When you return to App Center, refresh the browser to see your new profile photo. If your new profile photo doesn't immediately appear, hard refresh your browser to replace the previously cached image.

### Email Notifications

Manage your email preferences to sign up for automatic notifications for builds, distributions, and crashes.

### Bug Tracker Integration

Integrate bug trackers like Jira, Visual Studio Team Services (VSTS), Azure DevOps, and GitHub to stay informed when your app crashes.

### Webhooks

Create and enable webhooks to integrate with third-party applications you already use. Webhooks are a simple way to notify third-party applications when a specified event has occurred. The main goal of webhooks is to communicate important information from App Center to users rather than having users come to the portal, or run API calls to be notified when certain events happen.

App Center's webhooks allow users to send automatic notifications to connected applications for the following events:

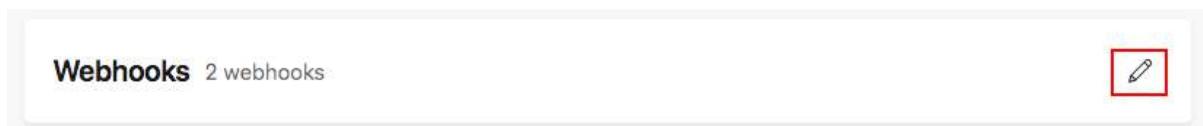
- Build:
  - Build success:
    - **Always:** when your app builds successfully
    - **Only if previously failed:** when your app has successfully built after one or more failed builds

- **Never**: you won't receive notifications for build success
- Build failure:
  - **Always**: when your app fails to build
  - **Only if previously successful**: when your app has failed to build after one or more successful builds
  - **Never**: you won't receive notifications for build failure
- Crashes: when a crash group is created
- Distribute: when a new version is released to a distribution group

App Center will send an HTTP POST payload to the webhook's specified URL. Webhooks are configured at the app level under the **Settings** page of your specified app. Users must have manager or developer permissions in the app to create and configure the webhooks. We currently only support webhooks for Slack and Microsoft Teams. To post to other platforms, you may write an Azure function that translates the way we POST to fit the requirements of the platform.

## Getting Started

1. Navigate to App Center, and select the specific app you want for webhooks integration.
2. In your app, in the far left-hand panel, select **Settings**
3. In the row panel titled **Webhooks**, go to the right-hand corner and click on the **pencil icon**, which brings up the **Webhooks** panel.



4. In the top-right corner, click the blue **New Webhook** button and enter:
  - Webhook name
  - Webhook URL

You can obtain the webhook URL from your integrated application's settings (for example, here are details on how to obtain the webhook URL from Microsoft Teams and how to obtain the webhook URL from Slack).

Select the **dropdown** for Build status notifications and the **checkbox** for Crashes and Distribute notifications to decide what events will trigger the webhook alerts.



5. Done! Your webhook is now created and enabled. You may create multiple webhooks by repeating step 4.
6. Toggle to the extreme right hand of the webhook to **test**, **disable**, or **delete** the webhook.
  - **test** will send a test alert to your connected application.
  - **disable** keeps the webhook inactive but present in your dashboard.
  - **delete** will remove the webhook from your dashboard.

When these events happen, App Center notifications are posted into your integrated applications. For example, here is how a Build success notification looks like with a connected Slack application:

App Center APP 6:14 PM  
DiceOut (Android)  
Build #2 succeeded. 🚀  
-  
Duration 1 min 15 secs Branch ErrorFreeBranch  
View details

## Example webhook payload

Here are examples of the JSON webhook payload for:

1. Build

```
{
  "app_name": "myFirstApp",
  "branch": "main",
  "build_status": "Succeeded",
  "build_id": "33",
  "build_link": "https://appcenter.ms/users/{user-id}/apps/{app-name}/build/branches/main/builds/33",
  "build_reason": "manual",
  "finish_time": "2018-06-14T23:59:05.2542221Z",
  "icon_link": "https://appcenter-filemanagement-distrib4ede6f06e.azureedge.net/f7794e4c-42f1-4e7c-8013-07ed2e1b733d/icon_1aunc_her.png?sv=2020-02-18&sr=c&sig=gs4JfcWjpKeYH%2F%2Fg0jEtSKKbeRkug9q%2FldslmzzeOg0%3D&se=2020-02-26T08%3A57%3A58Z&sp=r",
  "notification_settings_link": "https://appcenter.ms/users/{user-id}/apps/{app-name}/settings/notifications",
  "os": "iOS",
  "start_time": "2018-06-14T23:57:03.4379381Z",
  "source_version": "55820a357ba26831f2eeb3be9973a4ef20618b73",
  "sent_at": "2018-06-14T23:59:08.4897604Z"
}
```

## 2. Crash

```
{
  "id": "3698593379u",
  "name": "android.app.Activity.performResume (Activity.java:5084)",
  "reason": "android.app.SuperNotCalledException",
  "url": "https://appcenter.ms/orgs/{org-id}/apps/{app-name}/crashes/errors/3698273379u",
  "app_display_name": "{app-name}",
  "app_platform": "Java",
```

```

    "app_version": "2.0.1(42)",
    "stack_trace": [],
    "affected_users": 0,
    "crash_count": 0,
    "sent_at": "2019-05-16T23:47:31.4881512Z",
    "app_id": "48573473-f069-4715-8bab-9ae42cec48b2"
}

```

### 3. Distribute

```
{
    "app_name": "{app-name}",
    "app_display_name": "{app-display-name}",
    "release_id": "123",
    "platform": "Android",
    "uploaded_at": "2018-07-17T20:46:14Z",
    "fingerprint": "0abed1269e4ae3bf524e4cc7165f4f34",
    "release_notes": "",
    "version": "74",
    "short_version": "1.7.0",
    "min_os": "4.0.3",
    "mandatory_update": true,
    "size": 2634279,
    "provisioning_profile_name": null,
    "provisioning_profile_type": null,
    "bundle_identifier": "com.microsoft.appcenter.test",
    "install_link": "https://install.appcenter.ms/orgs/{org-name}/apps/{app-name}/releases/123?source=email",
    "icon_link": "https://appcenter-filemanagement-distrib4ede6f06e.azureedge.net/f7794e4c-42f1-4e7c-8013-07ed2e1b733d/icon_launcher.png"
}
```

```
sv=2020-02-18&sr=c&sig=gs4JfcWjpKeYH%2F%2Fg0jEtSKKbeRkug9q%2Fldslm  
zzeOg0%3D&se=2020-02-26T08%3A57%3A58Z&sp=r",  
    "distribution_group_id": "1a5a0605-4b9c-4de2-9a35-t569456df0cc",  
    "installable": true,  
    "sent_at": "2019-05-16T23:20:08.7799314Z",  
    "app_id": "f37c6194-9ac9-4504-be61-55re334r5649"  
}
```

## Slack App

Install the App Center Slack app to easily trigger builds, monitor your app, and invite new testers directly from Slack.