Overview - What is Azure Logic Apps?

03/11/2020 • 9 minutes to read • +1

In this article

Azure Logic Apps is a cloud service that helps you schedule, automate, and orchestrate tasks, business processes, and workflows when you need to integrate apps, data, systems, and services across enterprises or organizations. Logic Apps simplifies how you design and build scalable solutions for app integration, data integration, system integration, enterprise application integration (EAI), and business-to-business (B2B) communication, whether in the cloud, on premises, or both.

For example, here are just a few workloads you can automate with logic apps:

- Process and route orders across on-premises systems and cloud services.
- Send email notifications with Office 365 when events happen in various systems, apps, and services.
- Move uploaded files from an SFTP or FTP server to Azure Storage.
- Monitor tweets for a specific subject, analyze the sentiment, and create alerts or tasks for items that need review.

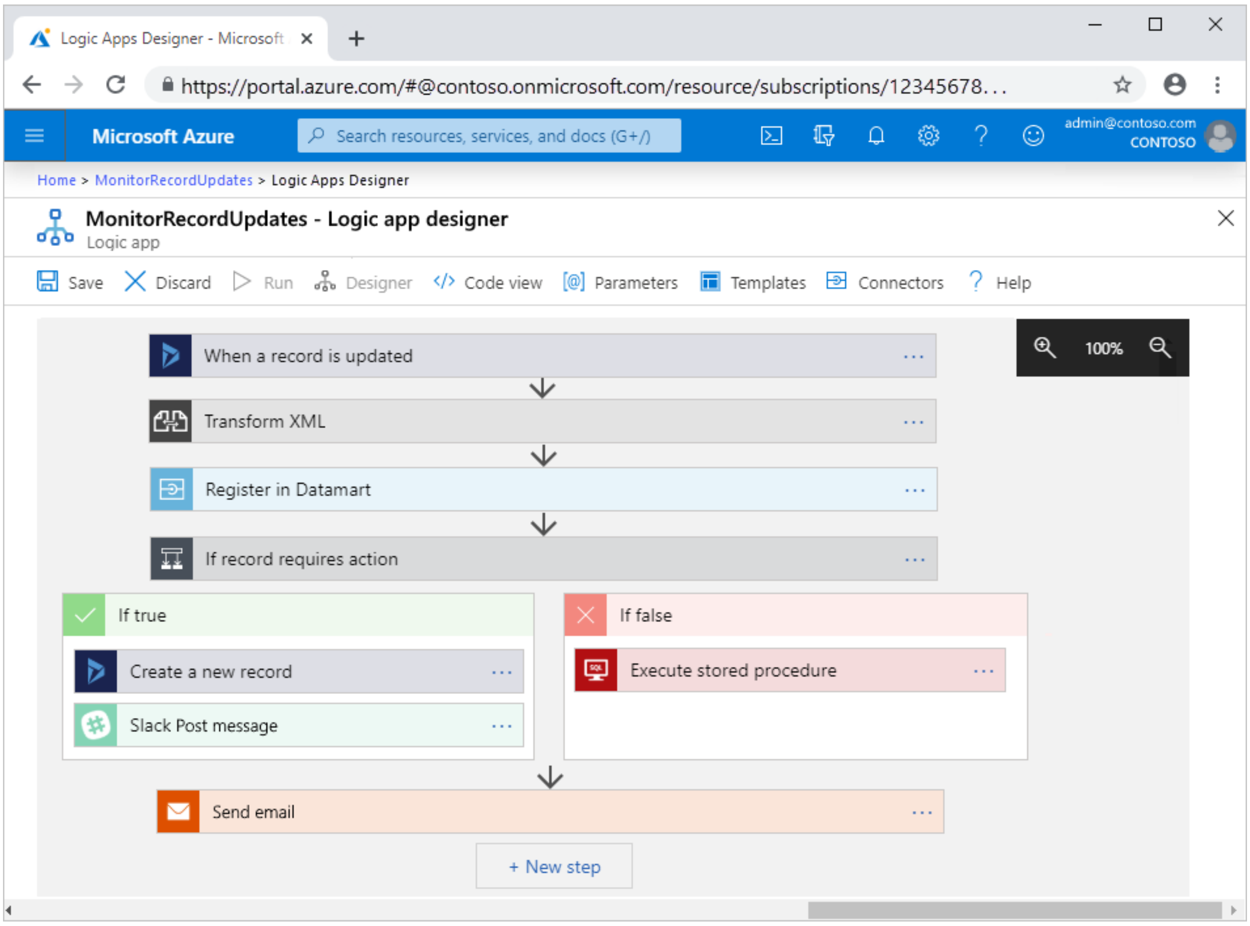
To build enterprise integration solutions with Azure Logic Apps, you can choose from a growing gallery with hundreds of ready-to-use connectors, which include services such as Azure Service Bus, Azure Functions, Azure Storage, SQL Server, Office 365, Dynamics, Salesforce, BizTalk, SAP, Oracle DB, file shares, and more. Connectors provide triggers, actions, or both for creating logic apps that securely access and process data in real time.

01:28

How do logic apps work?

Every logic app workflow starts with a trigger, which fires when a specific event happens, or when new available data meets specific criteria. Many triggers provided by the connectors in Logic Apps include basic scheduling capabilities so that you can set up how regularly your workloads run. For more complex scheduling or advanced recurrences, you can use a Recurrence trigger as the first step in any workflow. Learn more about schedule-based workflows.

Each time that the trigger fires, the Logic Apps engine creates a logic app instance that runs the actions in the workflow. These actions can also include data conversions and workflow controls, such as conditional statements, switch statements, loops, and branching. For example, this logic app starts with a Dynamics 365 trigger with the built-in criteria "When a record is updated". If the trigger detects an event that matches this criteria, the trigger fires and runs the workflow's actions. Here, these actions include XML transformation, data updates, decision branching, and email notifications.


Logic Apps Designer - Microsoft
https://portal.azure.com/#@contoso.onmicrosoft.com/resource/subscriptions/12345678...
Microsoft Azure
Search resources, services, and docs (G+/)
admin@contoso.com
CONTOSO
Home > MonitorRecordUpdates > Logic Apps Designer
MonitorRecordUpdates - Logic app designer
Logic app
Save
Discard
Run
Designer
Code view
Parameters
Templates
Connectors
Help
100%
When a record is updated
Transform XML
Register in Datamart
If record requires action
If true
Create a new record
Slack Post message
If false
Execute stored procedure
Send email
+ New step

You can build your logic apps visually with the Logic Apps Designer, which is available in the Azure portal through your browser and in Visual Studio. For more custom logic apps, you can create or edit logic app definitions in JavaScript Object Notation (JSON) by working in the "code view" editor. You can also use Azure PowerShell commands and Azure Resource Manager templates for select tasks. Logic apps deploy and run in the cloud on Azure. For a more detailed introduction, watch this video: Use Azure Enterprise Integration Services to run cloud apps at scale

Why use logic apps?

With businesses moving toward digitization, logic apps help you connect legacy, modern, and cutting-edge systems more easily and quickly by providing prebuilt APIs as Microsoft-managed connectors. That way, you can focus on your apps' business logic and functionality. You don't have to worry about building, hosting, scaling, managing, maintaining, and monitoring your apps. Logic Apps handles these concerns for you. Plus, you pay only for what you use based on a consumption pricing model.

In many cases, you won't have to write code. But if you must write some code, you can create code snippets with Azure Functions and run that code on-demand from logic apps. Also, if your logic apps need to interact with events from Azure services, custom apps, or other solutions, you can use Azure Event Grid with your logic apps for event monitoring, routing, and publishing.

Logic Apps, Functions, and Event Grid are fully managed by Microsoft Azure, which frees you from worries about building, hosting, scaling, managing, monitoring, and maintaining your solutions. With the capability to create "serverless" apps and solutions, you can just focus on the business logic. These services automatically scale to meet your needs, make integrations faster, and help you build robust cloud apps with minimal code.

To learn how companies improved their agility and increased focus on their core businesses when they combined Logic Apps with other Azure services and Microsoft products, check out these customer stories.

Here are more details about the capabilities and benefits that you get with Logic Apps:

Visually build workflows with easy-to-use tools

Save time and simplify complex processes with visual design tools. Build logic apps from start-to-finish by using the Logic Apps Designer through your browser in the Azure portal or in Visual Studio. Start your workflow with a trigger, and add any number of actions from the connectors gallery.

Get started faster with logic app templates

Create commonly used solutions more quickly when you choose predefined workflows from the template gallery. Templates range from simple connectivity for software-as-a-service (SaaS) apps to advanced B2B solutions plus "just for fun" templates. Learn how to create logic apps from prebuilt templates.

Connect disparate systems across different environments

Some patterns and workflows are easy to describe but hard to implement in code. Logic apps help you seamlessly connect disparate systems across on-premises and cloud environments. For example, you can connect a cloud marketing solution to an on-premises billing system, or centralize messaging across APIs and systems with an Enterprise Service Bus. Logic apps provide a fast, reliable, and consistent way to deliver reusable and reconfigurable solutions for these scenarios.

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, and X12. With the features in the Enterprise Integration Pack (EIP), you can build logic apps that transform message formats used by your partners into formats that your organization's systems can interpret and process. Logic Apps handles these exchanges smoothly and also securely with encryption and digital signatures.

Start small with your current systems and services, and grow incrementally at your own pace. When you're ready, Logic Apps and the EIP help you implement and scale up to more mature integration scenarios by providing these capabilities and more:

- Build off these products and services:
  - Microsoft BizTalk Server
  - Azure Service Bus
  - Azure Functions
  - Azure API Management
- Process XML messages
- Process flat files
- Exchange messages with EDIFACT, AS2, and X12 protocols
- Store and manage these B2B artifacts and more in one place with integration accounts:
  - Partners
  - Agreements
  - XML transform maps
  - XML validation schemas

For example, if you're using Microsoft BizTalk Server, logic apps can communicate with your BizTalk Server by using the BizTalk Server connector. You can then extend or perform BizTalk-like operations in your logic apps by including integration account connectors, which are available with the Enterprise Integration Pack.

Going in the other direction, BizTalk Server can connect to and communicate with logic apps by using the Microsoft BizTalk Server Adapter for Logic Apps. Learn how to set up and use the BizTalk Server Adapter in your BizTalk Server.

Write once, reuse often

Create your logic apps as Azure Resource Manager templates so that you can automate logic app deployment across multiple environments and regions.

Access resources inside Azure virtual networks

Logic apps 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 Logic Apps service that uses dedicated resources and runs separately from the "global" multi-tenant Logic Apps service.

Running logic apps in your own separate dedicated instance helps reduce the impact that other Azure tenants might have on your apps' 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 additional 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, see 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, see Connect to Azure virtual networks from Azure Logic Apps.

Built-in extensibility

If you don't find the connector that you want to run custom code, you can extend logic apps by creating and calling your own code snippets on-demand through Azure Functions. Create your own APIs and custom connectors that you can call from logic apps.

Pay only for what you use

Logic Apps uses consumption-based pricing and metering unless you have logic apps previously created with App Service plans.

Learn more about Logic Apps with these introductory videos:

- Integration with Logic Apps - Go from zero to hero
- Enterprise integration with Microsoft Azure Logic Apps

- Building advanced business processes with Logic Apps

How does Logic Apps differ from Functions, WebJobs, and Power Automate?

All these services help you "glue" and connect disparate systems together. 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, see Choose between Logic Apps, Functions, WebJobs, and Power Automate.

Key terms

- **Workflow**: Visualize, design, build, automate, and deploy business processes as series of steps.

- **Managed connectors**: Your logic apps need access to data, services, and systems. You can use prebuilt Microsoft-managed connectors that are designed to connect, access, and work with your data. See Connectors for Azure Logic Apps.

- **Triggers**: Many Microsoft-managed connectors provide triggers that fire when events or new data meet specified conditions. For example, an event might be getting an email or detecting changes in your Azure Storage account. Each time the trigger fires, the Logic Apps engine creates a new logic app instance that runs the workflow.

- **Actions**: Actions are all the steps that happen after the trigger. Each action usually maps to an operation that's defined by a managed connector, custom API, or custom connector.

- **Enterprise Integration Pack**: For more advanced integration scenarios, Logic Apps includes capabilities from BizTalk Server. The Enterprise Integration Pack provides connectors that help logic apps easily perform validation, transformation, and more.

Get started

Logic Apps is one of the many services hosted on Microsoft Azure. So before you start, you need an Azure subscription. If you don't have a subscription, sign up for a free Azure account.

If you have an Azure subscription, try this quickstart to create your first logic app, which monitors new content on a website through an RSS feed and sends email when new content appears.

Next steps

- Check traffic with a schedule-based logic app
- Learn more about serverless solutions with Azure
- Learn more about B2B integration with the Enterprise Integration Pack

Is this page helpful?

👍 Yes 👎 No