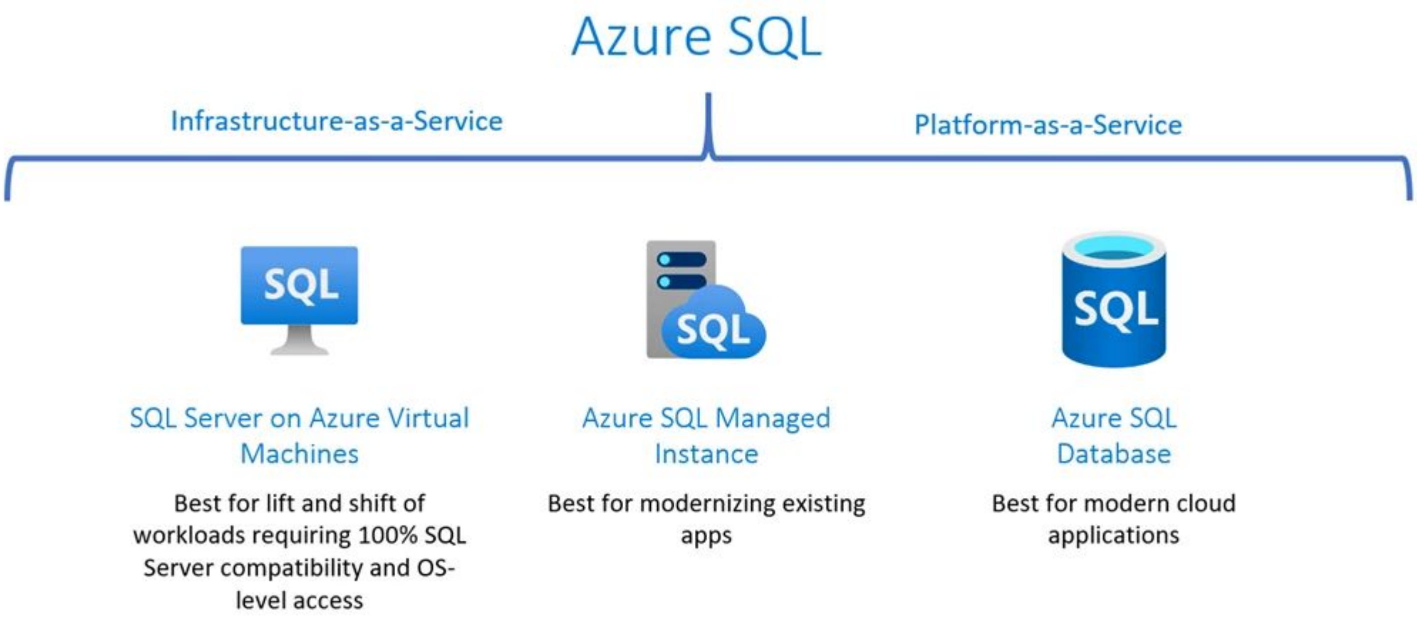
Azure SQL Product Family

What is Azure SQL?

Azure SQL is a family of managed, secure, and intelligent products that use the SQL Server database engine in the Azure cloud. Azure SQL is built upon the familiar SQL Server engine.



SQL Server on Azure Virtual Machines

Best for lift and shift of workloads requiring 100% SQL Server compatibility and OS-level access

Azure SQL Managed Instance

Best for modernizing existing apps

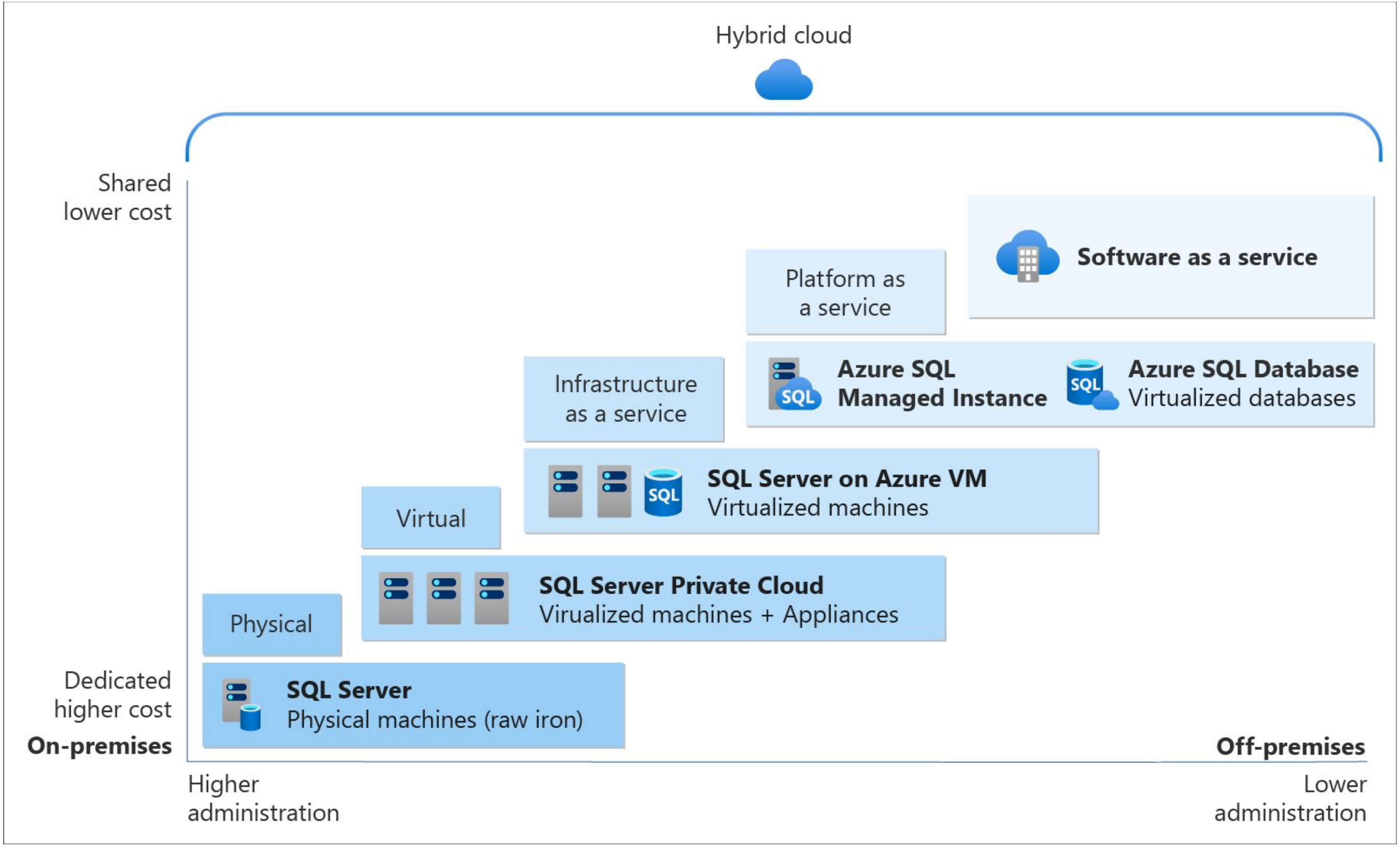
Azure SQL Database

Best for modern cloud applications

Products in Azure SQL Family

The three products in the Azure SQL family are:

- Azure SQL Database: Support modern cloud applications on an intelligent, managed database service that includes serverless compute.
- Azure SQL Managed Instance: Modernize your existing SQL Server applications at scale with an intelligent fully managed instance as a service, with almost 100% feature parity with the SQL Server database engine. Best for most migrations to the cloud.
- SQL Server on Azure VMs: Lift-and-shift your SQL Server workloads with ease and maintain 100% SQL Server compatibility and operating system-level access.


Hybrid cloud
Shared lower cost
Dedicated higher cost
On-premises
Off-premises
Higher administration
Lower administration
Physical
SQL Server
Physical machines (raw iron)
Virtual
SQL Server Private Cloud
Virualized machines + Appliances
Infrastructure as a service
SQL Server on Azure VM
Virtualized machines
Platform as a service
SQL
Azure SQL Managed Instance
SQL
Azure SQL Database
Virtualized databases
Software as a service

As seen in the diagram, each service offering can be characterized by the level of administration you have over the infrastructure, and by the degree of cost efficiency.

In Azure, you can have your SQL Server workloads running as a hosted service (PaaS), or a hosted infrastructure (IaaS) supporting the software layer, such as Software-as-a-Service (SaaS) or an application.

Within PaaS, you have multiple product options, and service tiers within each option.

The key question that you need to ask when deciding between PaaS or IaaS is - do you want to manage your database, apply patches, and take backups - or do you want to delegate these operations to Azure?

Azure SQL Database

Azure SQL Database is a fully-managed relational database service provided by Microsoft Azure. It offers high availability, security, and scalability, making it ideal for cloud-based applications.

Key Features:

- Fully managed database as a service (DBaaS).
- Built-in high availability with automated backups.
- Supports advanced security features like encryption and auditing.

Database Models and Options in Azure SQL

Azure SQL Database comes with different deployment models and options that suit various use cases.

- **Single Database**: Independent databases optimized for small to medium-sized applications.
- **Elastic Pool**: A collection of databases that share resources. Ideal for cost optimization when running multiple databases.
- **Managed Instance**: Provides almost full compatibility with on-prem SQL Server. Useful for migrating legacy SQL Server applications to the cloud.

Key Components of Azure SQL Database

Compute Tier: Defines how much processing power your database has.

- **Provisioned**: Fixed resources dedicated to the database.
- **Serverless**: Dynamically scales based on demand, ideal for workloads that experience variable traffic.

Storage Tier: The capacity you allocate for data storage. Azure SQL Database supports different performance tiers depending on the workload (Basic, Standard, and Premium tiers).

Security in Azure SQL Database

Azure SQL Database ensures security at various levels:

- **Transparent Data Encryption (TDE)**: Encrypts the database, logs, and backups.
- **Azure Active Directory (AAD)**: Provides centralized identity management and single sign-on for database users.
- **Firewall Rules**: Allows control over which IP addresses can access the database.

High Availability and Disaster Recovery

Azure SQL Database provides automatic backups and geo-replication to ensure availability and business continuity.

- **Geo-Replication**: Allows you to replicate your database to different Azure regions for disaster recovery.
- **Automatic Failover**: Ensures minimal downtime by switching to a replica in case of a database failure.

Scaling Azure SQL Database

Azure SQL Database can scale vertically (increasing resources for a single database) or horizontally (distributing databases across servers). This is useful for handling varying workloads.

- **Vertical Scaling**: Increase the compute power, memory, or storage.
- **Horizontal Scaling**: Distribute load across multiple databases using Elastic Pools.

Monitoring and Performance Tuning

Azure SQL Database comes with built-in monitoring and diagnostic tools.

- **Query Performance Insight**: Helps identify long-running queries and optimize database performance.
- **Automatic Tuning**: Detects and applies performance improvements automatically.

Backup and Restore

Azure SQL Database automatically takes backups to ensure data protection.

- **Automated Backups**: Full, differential, and transaction log backups are taken automatically and stored for up to 35 days.
- **Point-in-Time Restore**: Allows you to restore a database to any point within the backup retention period.

Connecting to Azure SQL DB

You can connect to Azure SQL Database using different tools:

- **Azure Data Studio**: A lightweight editor for SQL development.
- **SQL Server Management Studio (SSMS)**: A comprehensive management tool for SQL databases.
- **Azure Portal**: Provides a web-based interface for managing databases.

Use Cases

Web Applications: Azure SQL Database is a great choice for hosting data for web apps.

E-Commerce: Handles transactional workloads and scales easily for high-traffic e-commerce sites.

Data Analytics: Combined with other Azure services, SQL Database can be used for big data analysis.

Lab

- Create an Azure SQL Database.
- Configure the database for access.
- Connect the database to SSMS.
- Explore the database using basic SQL queries.

Serverless Technology

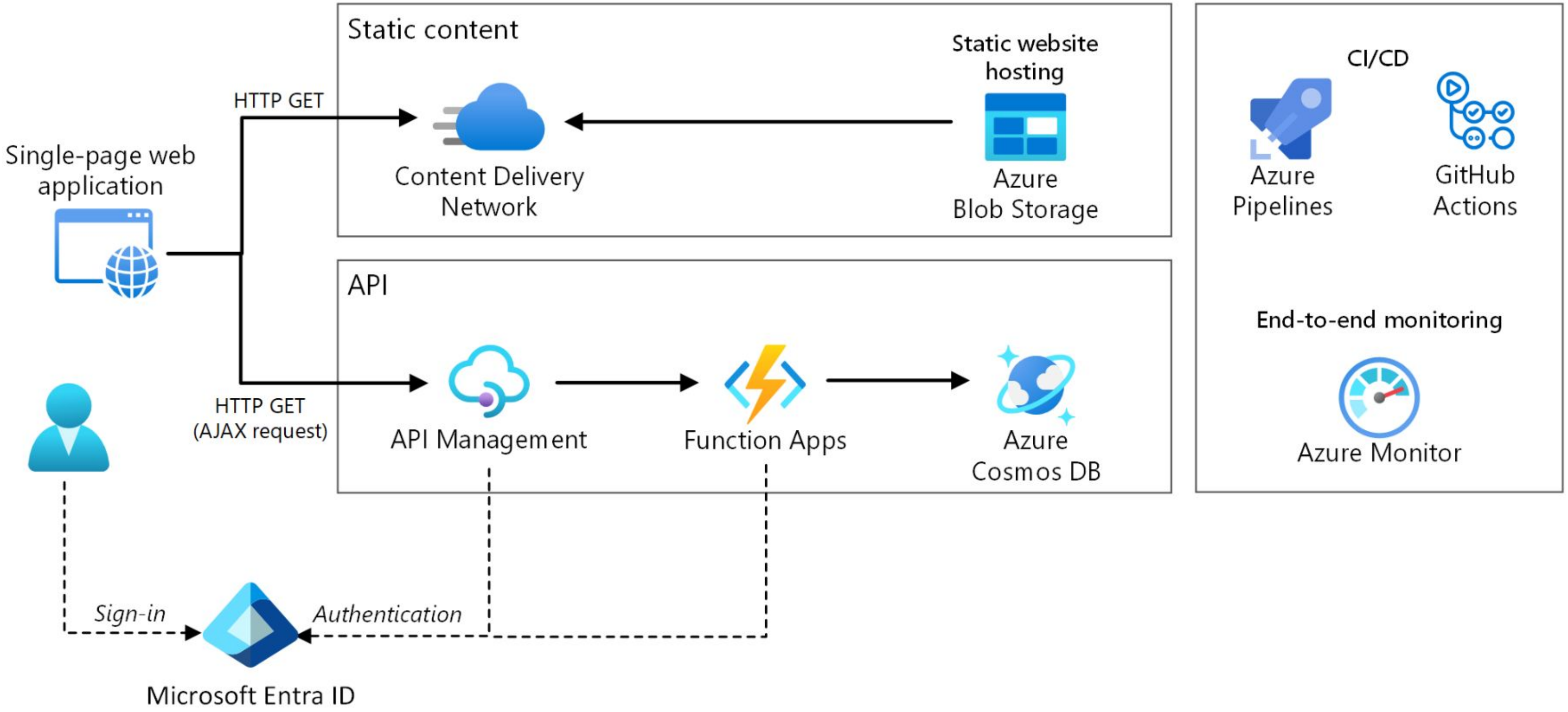
Introduction to Serverless Technology in Azure

Serverless computing allows developers to build and deploy applications without worrying about managing the underlying infrastructure.

This approach means the cloud provider takes care of server management, scaling, and maintenance, while you focus on writing code and building functionality.

In Azure, three key services that support serverless technology are **Azure Logic Apps**, **Azure Functions**, and **Azure Event Grid**.

These services allow seamless integration, automation, and event-driven workflows for cloud applications.



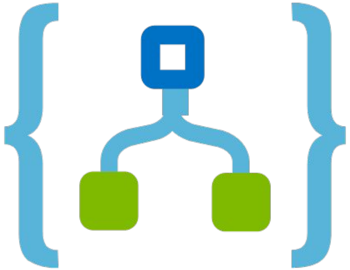
Microsoft Azure

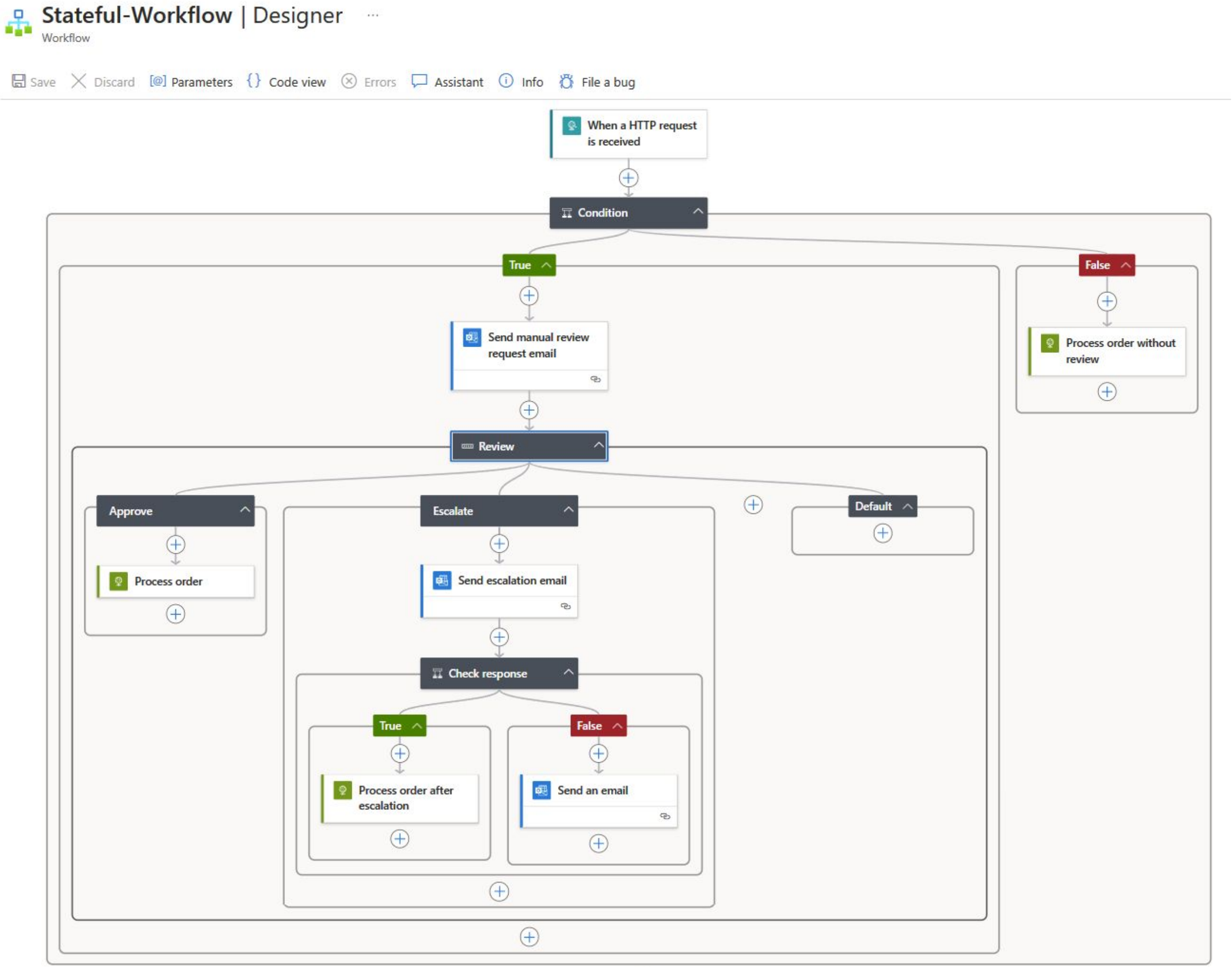
Azure Logic Apps

Azure Logic Apps is a cloud-based service that enables the automation and orchestration of workflows.

It allows you to design workflows that integrate various services and applications without needing to write extensive code.

This makes it perfect for automating repetitive tasks or connecting systems in a serverless environment.





https://learn.microsoft.com/en-us/azure/logic-apps/logic-apps-overview

Key Features

Workflow Automation: Automates tasks and business processes.

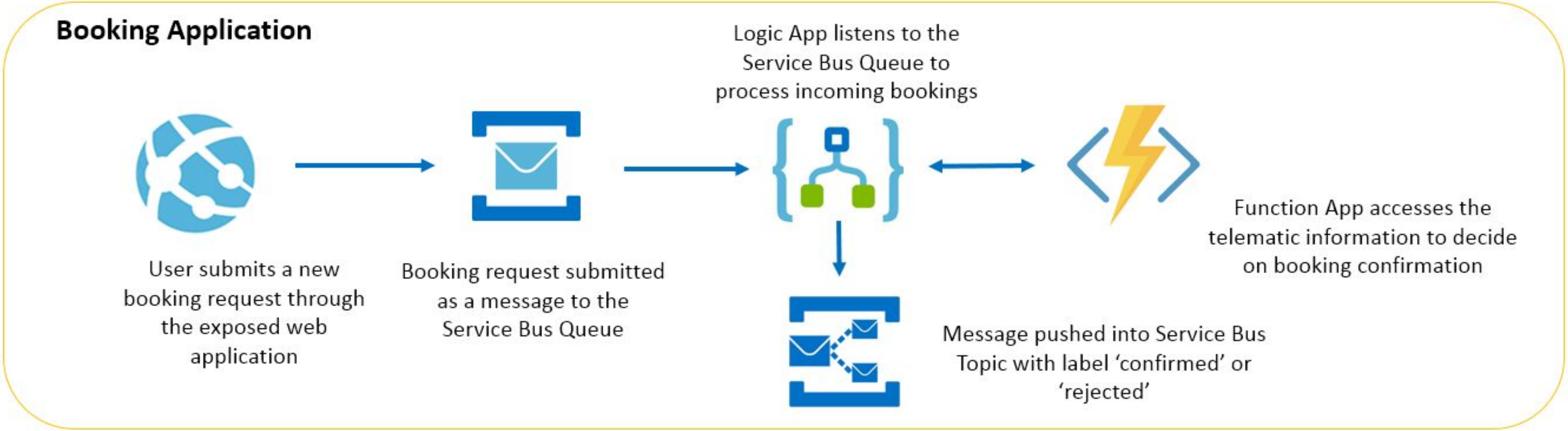
Integration with Services: Logic Apps can connect with hundreds of services, including SaaS applications (e.g., Office 365, Salesforce) and on-premises systems.

Visual Designer: A drag-and-drop interface that simplifies the process of creating workflows.

Event-Driven: Logic Apps can trigger actions based on specific events, such as receiving an email or a database update.

Use Cases

- **Automating Data Flow**: Extracting data from one system, transforming it, and loading it into another system.
- **Business Process Automation**: Automating tasks like sending approvals or managing order workflows.
- **Integrating Cloud and On-Premises**: Bridging systems that run in different environments.



Booking Application

User submits a new booking request through the exposed web application

Booking request submitted as a message to the Service Bus Queue

Logic App listens to the Service Bus Queue to process incoming bookings

Function App accesses the telematic information to decide on booking confirmation

Message pushed into Service Bus Topic with label ‘confirmed’ or ‘rejected’

Azure Functions

Azure Functions is a serverless compute service that allows you to run code in response to various triggers, such as HTTP requests, timers, or messages from Azure services. It's designed to execute small pieces of code (functions) on demand and scales automatically based on the number of incoming requests.

Key Features

Event-Driven Execution: Functions can be triggered by events such as HTTP requests, message queues, and file uploads.

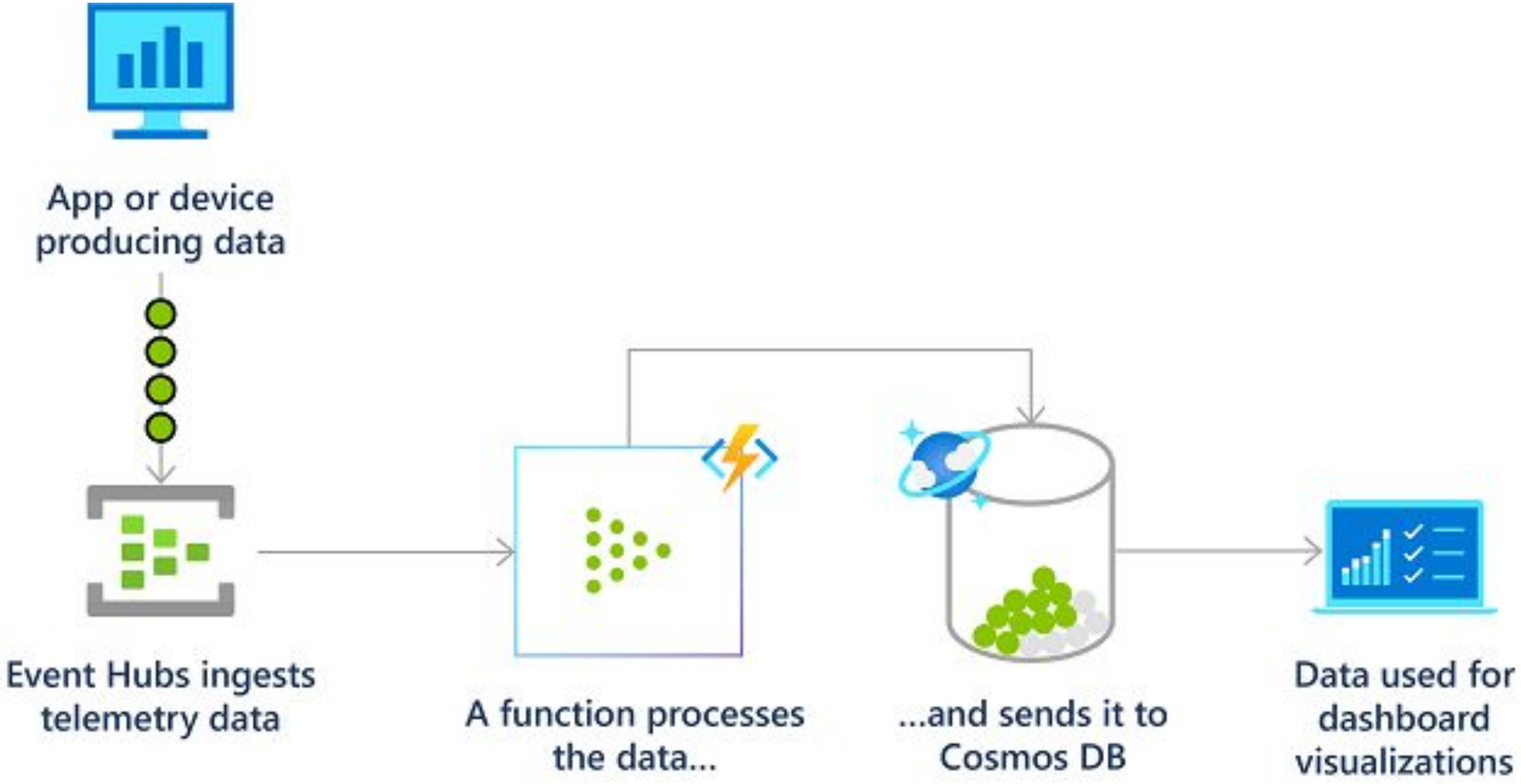
Automatic Scaling: Azure Functions scales up or down automatically depending on the demand, ensuring efficient use of resources.

Supports Multiple Languages: You can write functions in various programming languages, such as C#, JavaScript, Python, and PowerShell.

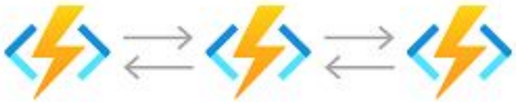
Cost-Effective: You only pay for the execution time of your functions, making it a highly cost-effective solution for running small tasks.

Use Cases

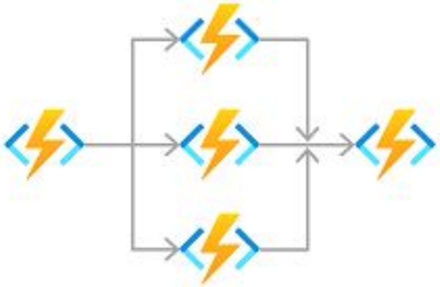
- **Real-Time Data Processing**: Handling real-time streams of data, such as IoT device data or web traffic logs.
- **Scheduled Tasks**: Running functions at scheduled intervals, such as daily backups or cleanups.
- **Microservices Architecture**: Creating isolated, independent services that perform specific tasks as part of a broader application


App or device producing data
Event Hubs ingests telemetry data
A function processes the data...
...and sends it to Cosmos DB
Data used for dashboard visualizations

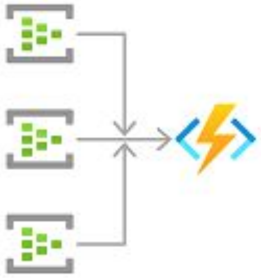
https://learn.microsoft.com/en-us/azure/azure-functions/functions-scenarios?pivots=programming-language-csharp



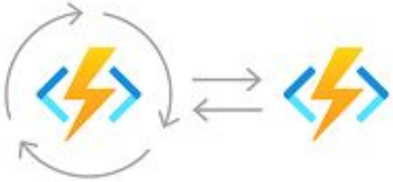
Manageable sequencing + error handling/compensation



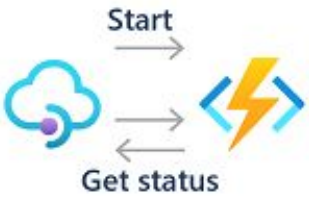
Fanning out and fanning in



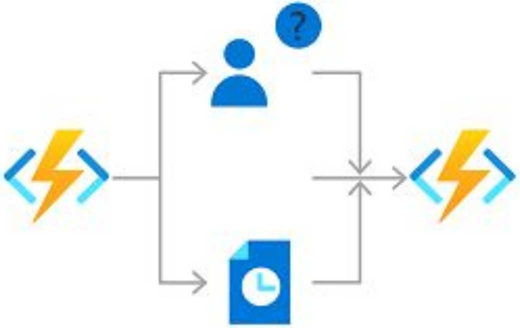
External events correlation



Flexible automated long-running process monitoring



Http-based async long-running APIs



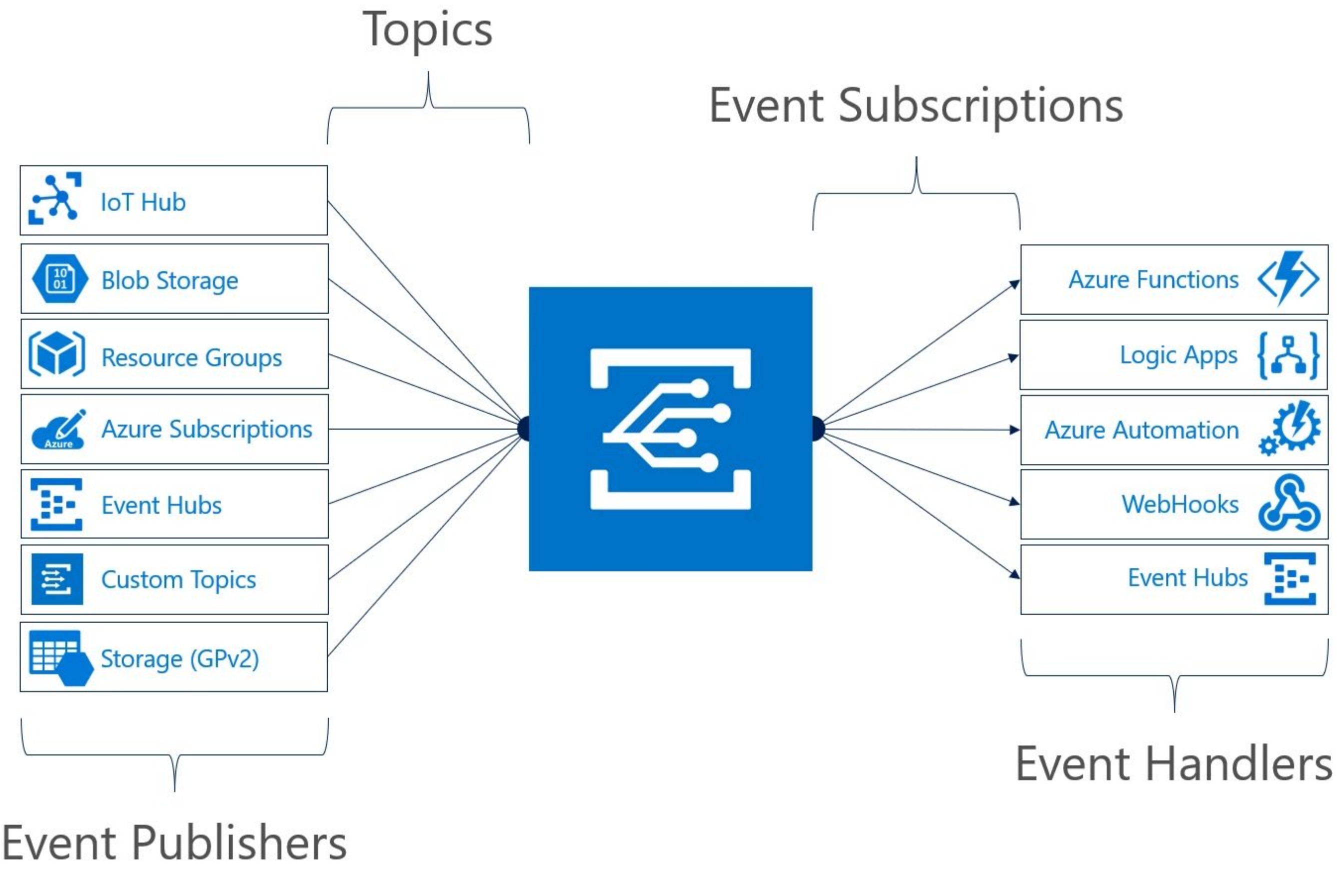
Human interaction

https://learn.microsoft.com/en-us/azure/azure-functions/functions-scenarios?pivots=programming-language-csharp

Azure Event Grid

Azure Event Grid is a fully managed event routing service that allows different applications and services to communicate with each other through events.

It’s designed to enable event-driven architectures where components of the system respond to changes and activities in real-time.



https://azure.microsoft.com/en-us/blog/announcing-the-general-availability-of-azure-event-grid/

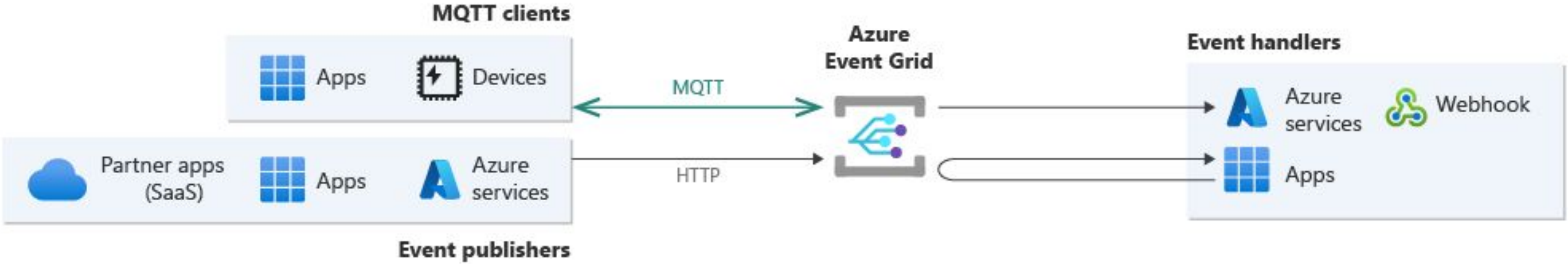
Key Features

Event Routing: Event Grid routes events from various sources (like Azure Blob Storage, Functions, or custom sources) to destinations (such as Logic Apps, Functions, or external services).

Real-Time Event Handling: Allows for immediate reaction to events as they happen.

Decoupled Architecture: Services can remain independent of each other, connected only through the events they generate and consume.

Scalable: Built to handle millions of events per second with minimal latency

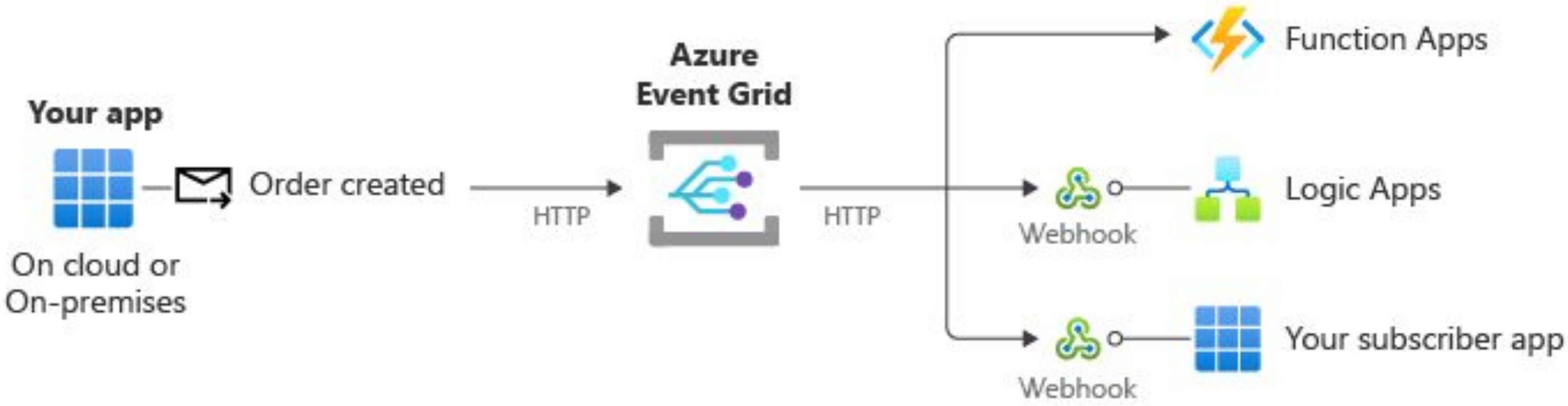


https://learn.microsoft.com/en-us/azure/event-grid/overview

Use Cases

- **Monitoring and Alerts**: Automatically triggering alerts or workflows when specific events happen, such as the creation of new files in storage.
- **Automating Responses**: Automatically starting processes like Azure Functions or Logic Apps when a particular event occurs.
- **Decoupled Systems**: Connecting different systems or microservices by using event-driven communication without hard dependencies.

Receive events from your applications

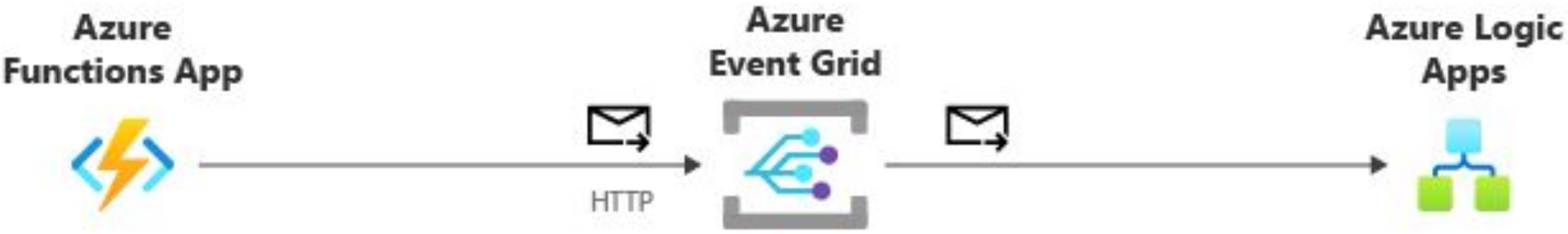


https://learn.microsoft.com/en-us/azure/event-grid/overview

How These Services Work Together

- **Event Grid** can route events to trigger **Azure Functions**, which can process the event and perform specific actions, such as transforming data or updating records.
- **Azure Logic Apps** can also be triggered by events in **Event Grid**, automating workflows like sending notifications or calling APIs when an event occurs.
- Together, these services enable **real-time, event-driven architectures** with minimal infrastructure management, allowing you to build scalable and responsive cloud applications

Build event-driven serverless solutions



https://learn.microsoft.com/en-us/azure/event-grid/overview

Conclusion

Azure’s serverless services— **Logic Apps**, **Functions**, and **Event Grid**—offer powerful tools for automating processes, integrating systems, and building event-driven applications.

They allow developers to focus on writing business logic without worrying about underlying infrastructure, making cloud computing more efficient, scalable, and cost-effective.

In cloud computing, understanding and utilizing these tools will enable rapid development and deployment of modern, serverless applications.

Labs

- Create an Azure Function that responds to HTTP requests by returning a message.
- Create an Azure Logic App that sends an email when a new file is added to a specific folder in OneDrive.
- Install the following:
    - Azure Functions Core Tools - https://learn.microsoft.com/en-us/azure/azure-functions/functions-run-local?tabs=windows%2Cisolated-process%2Cnode-v4%2Cpython-v2%2Chttp-trigger%2Ccontainer-apps&pivots=programming-language-csharp#v2
    - Azure Functions extension on VS code.