

How Azure SQL Edge works

8 minutes

In this unit, you learn how Azure SQL Edge works and whether it's suitable for your organization. Let's start by exploring how Azure SQL Edge helps address challenges that developers face when developing IoT apps.

Azure SQL Edge is a containerized Linux application that runs on a process that's based on ARM64 or x64 architecture. The Azure SQL Edge startup memory footprint is less than 500 megabytes (MB). Within this small footprint, you can design and build apps that run on many IoT devices.

💡 Tip

These devices can include battery-powered or solar-powered devices that you deploy in remote areas that have only occasional connectivity.

In this unit, you learn how Azure SQL Edge:

- Captures continuous streams of increasing data volumes in real time.
- Integrates data from the edge to cloud to on-premises.
- Supports several platforms.
- Provides excellent security and performance.

Capture continuous data streams in real time

Many organizations deal with large data volumes that they need to ingest and analyze in a timely manner, making it vital that any edge-based solution you select can capture continuous real-time streams of increasing amounts of data. It's also important that you can derive valuable insights about it. Azure SQL Edge supports a built-in streaming engine to help address these needs.

Time-series and streaming data support

IoT sensors typically transmit time-sequenced measurements, or *time-series data*, as a data stream. There are two aspects of the time-series capabilities offered by Azure SQL Edge:

- A built-in streaming engine that allows:
 - Transformation.
 - Windowed aggregation.
 - Simple anomaly detection.
 - Classification of incoming data streams.

ⓘ **Note**

This engine is built using the same codebase as Azure Stream Analytics.

- A time-series storage engine that allows storage of time-indexed data, that you can aggregate in the cloud for future analysis.

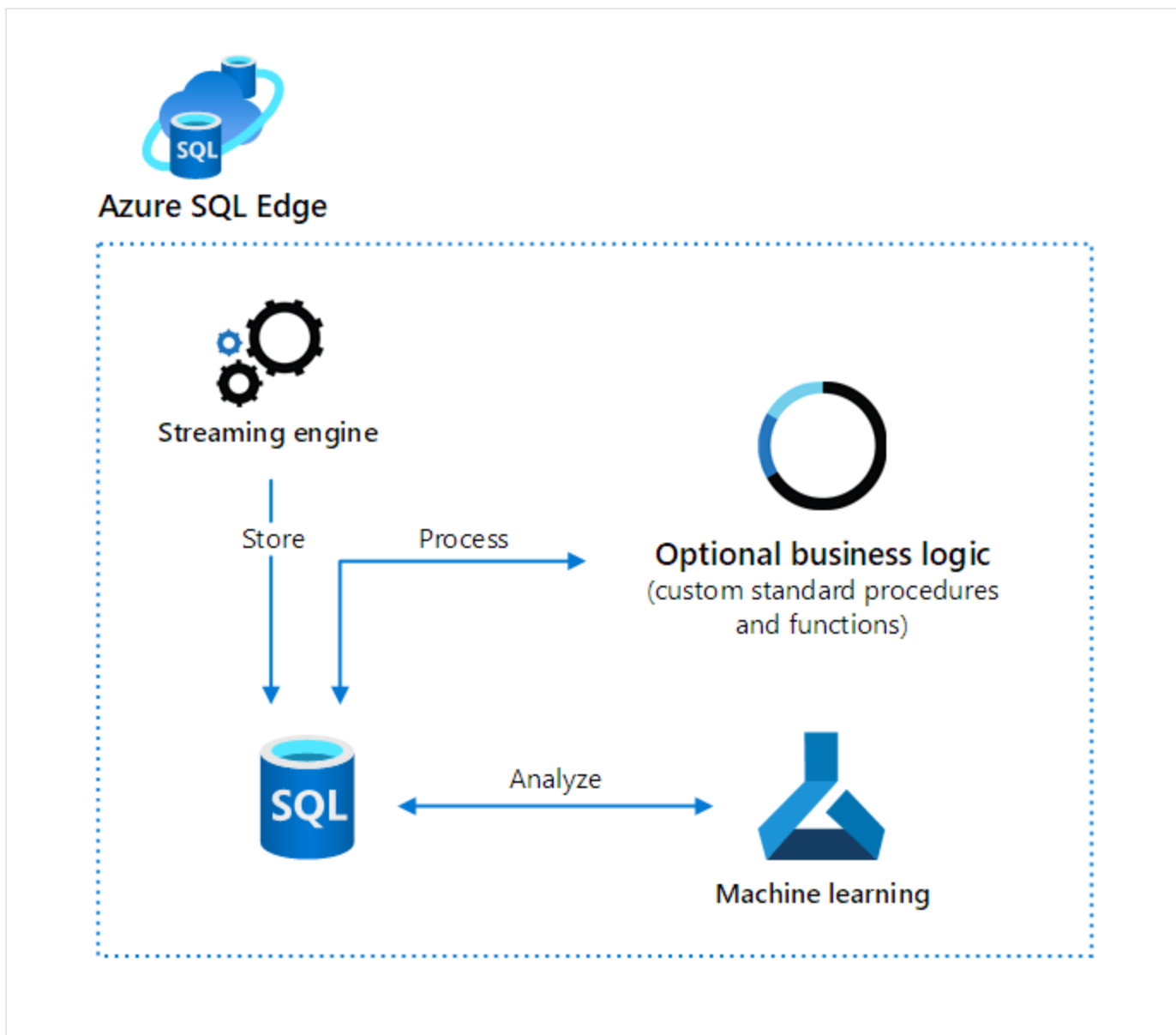
The following graphic depicts:

- The streaming data engine with native analytics.
- An SQL database engine that contains built-in machine-learning capabilities.

The graphic indicates that the local SQL database can aggregate the real-time streamed data that the streaming analytics engine ingests.

💡 **Tip**

You can process data by using standard procedures and functions, as necessary.



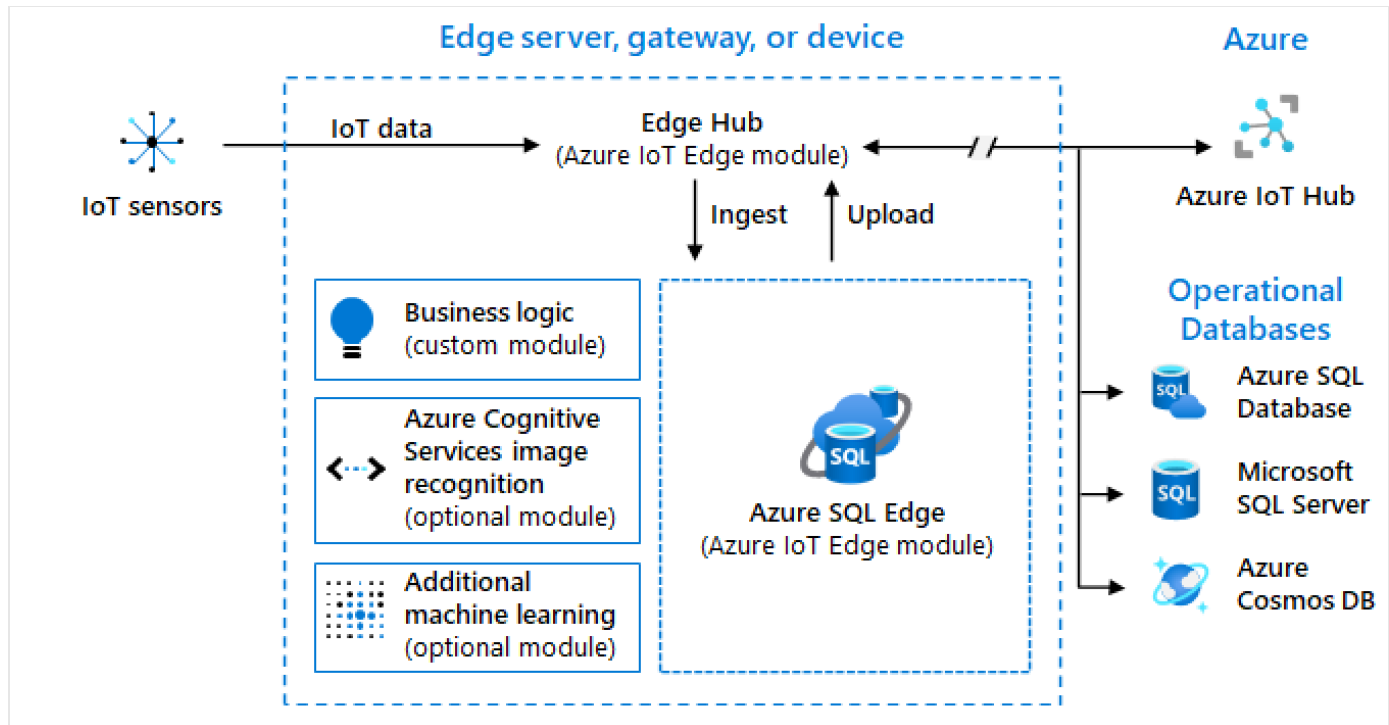
Integrate data

IoT app developers previously recognized that edge data couldn't always be easily integrated in a comprehensive organizational data solution. This lack of integration resulted in IoT device data at the edge being inaccessible in a proprietary solution. Azure SQL Edge eliminates the problem by using the same codebase as SQL Server, Azure SQL Database, and Azure SQL Managed Instance.

As the following graphic depicts, Azure SQL Edge offers compatibility with other Microsoft services and apps by providing:

- Synchronization with Azure services such as:
 - SQL Database and SQL Managed Instance.
 - SQL Server on an Azure virtual machine (VM).

- Azure Cosmos DB.
- The ability to choose data sources (sensors) and targets that best suit your organization's needs.
- Native integration with Azure products and services such as:
 - Azure IoT Edge.
 - Azure Stack Edge.



Run Azure SQL Edge on the appropriate platform

Edge scenarios must often run on different platforms. Azure SQL Edge provides you with the following options:

- **Architecture.** Device architecture supports architectures based on ARM64 and x64.
- **Container OS.** Choose either Windows or Linux as your container's host.
- **Container type.** Choose either Kubernetes or Linux Docker containers to orchestrate your device infrastructure for better efficiency and automation.
- **Mode.** Online or offline modes support various edge environments.

Protect data at the edge

Security is a primary concern when deploying IoT apps to the edge. Any device that's accessible over the internet is at risk of being a malicious hacker's target. So, any data stored on that device can be comprised.

SQL Server is the most secure database platform available. Because Azure SQL Edge is based on SQL Server technology, it has the same security features of SQL Server Enterprise. The following table details these built-in security features.

[Expand table](#)

Feature	Description
RBAC and ABAC	<i>Role-based access control</i> (RBAC) manages access to specific resources based on permissions assigned to the user requesting access (or a group to which they belong). <i>Attribute-based access control</i> (ABAC) manages access control based on rules that refer to the attributes of the user, target data, or resource.
Encryption	<i>Data protection with Transparent Data Encryption</i> (TDE) enables compliance with many security regulations, by encrypting database and log files through an encryption key that is encrypted. <i>Always Encrypted</i> provides separation between users who own the data and the managers of that data. This encryption helps prevent users with administrator privileges to a database from accessing encrypted data they don't own in that database.
Classification	A feature that helps you comply with security regulations by allowing you to categorize it by sensitivity and business impact. Once you classify data, you can manage it in ways that help protect sensitive or important data from theft or loss.

Next unit: When to use Azure SQL Edge

[Continue >](#)