# What to look out for when selecting edge computing solutions

**BOSCH**
Invented for life

An edge computing solution typically sits between the IoT devices, the gateways to connect the devices, and the IoT cloud platform. The basic features of an edge solution focus on connecting the devices, processing and managing the data locally, and moving data to the cloud platform for further data analytics and integration with back end services.

Many questions arise, when an organization is about to decide on an edge solution: How to select the right technology? How to optimize value? What to consider for a new project?

While each project comes with individual requirements, there is a universal set of features that should be considered when selecting an edge solution.

## Support for different edge platforms

The variety of available edge devices is changing rapidly. Future-proof edge solutions employ lightweight native components to run on a choice of platforms from small microcontrollers to powerful edge nodes.

## Rule engine

A simple rule engine should be available for the automatic execution of predefined business rules, local monitoring, and control of various aspects on the edge. Rules typically have the following form:
- ON - something happens (window state changed)
- IF - conditions are satisfied (window.state = open state)
- THEN - do the task (light.state = off)

## Local persistence storage

Edge solutions should allow for local storage of data on the gateway without requiring connectivity with the cloud. Different options for storage might include relational database, normalized telemetry data from edge resources and devices, and the ability to add your own storage types.

## Autonomous operation while disconnected

Edge solutions should be able to function without connectivity to the cloud and possess robust capabilities to handle cloud connection interruptions and ensure consistency and autonomous operation. The edge solution must manage cloud connectivity status, publish notifications on cloud connectivity changes, buffer messages during connection loss, manage connection retries, and synchronize messages on successful reconnection. This way, the IoT applications at the edge stay up and running even if the connection to the cloud is interrupted, unreliable, or lost.

## Local applications

It should be possible to develop local applications that are installed and executed at the edge.

## Support for device protocols

There is a wide variety of communication protocols used to connect sensors and actuators. To facilitate connectivity of these devices, an edge gateway should support prominent IoT protocols such as Z-Wave, ZigBee, KNX, DECT ULE, Bluetooth LE, EEBus SPINE, EEBus SHIP, HomeConnect, Modbus, UPnP, SNMP, PROFINET, wMBus, ONVIF, EnOcean, BACnet, OPC UA, LoRa, IP cameras and Siemens S7.

## Device abstraction and digital twin

To make it easier to create applications for an edge solution, a device abstraction layer should offer a unified interface for devices, regardless of the device type or connectivity protocol. The abstraction should support implementing applications without the need to know details of a specific protocol. A digital twin, controlled from the cloud, should be able to access the same device abstraction layer to monitor and control the devices in a consistent manner.

## Cloud connectivity out of the box

To simplify creating edge solutions, cloud connectivity should be available out of the box. Lightweight components that can run on even the smallest microcontrollers should enable the devices with IoT essentials such as cloud connectivity, local messaging, software management, and container management and integration with the cloud services platform.

## Availability and reliability

Edge computing solutions should be able to constantly

monitor the health of the gateway runtime, detect faults, and execute actions like reboot, restart, or initiate a factory reset. Additionally it should be possible to backup important user data on a periodic basis or with an explicit request.

## Remote management and update
It should be possible to remotely manage edge computing devices to start, stop, configure the device, and remotely install new software and hotfixes.

## Analytics and machine learning
An edge computing solution should make it possible to apply different machine learning algorithms at the edge. Advanced capabilities might include the ability to combine local solutions on the edge with algorithms and services on the cloud.

## Integration of custom applications
Custom applications or advanced edge services should be deployable at the edge as standard OCI containers. Containers allow creation of scalable applications using most fit-for-purpose technology and language, reuse of cloud applications without rewriting code, or seamless access to IoT device data and edge services.

## Security and privacy
Key security and privacy features of an edge solution should include:
- The ability to configure the permissions of components and applications based on location, signer, and other custom conditions.
- Management and configuration of security policies for controlling access to various gateway resources in an easy way.
- Support for secure communication to the gateway runtime through TLS and data encryption.
- Certificate management including revocation and update of certificates.
- Integration with 3rd party key management services, public key infrastructures, and hardware security modules.

## Creating the intelligent edge

Distributing intelligence to the edge requires enabling IoT devices with local computing, management, messaging and AI capabilities to realize a variety of business models and end-to-end scenarios.

**Bosch IoT Edge** provides the tools and services needed to connect any device to the cloud, set communication between devices, and develop IoT applications – across different platforms, languages, and system requirements. It allows you to build interoperable solutions, accelerate application development, scale your hardware and applications and manage security and updates.

- **Run on a choice of platforms:** Lightweight native components can be used on small microcontrollers or powerful edge nodes to scale your hardware.
- **Use fit-for-purpose language:** Standard containers allow you to selectively scale applications, reuse code, and access advanced edge functions.
- **Integrate quickly:** Native connectivity and an open MQTT/Ditto protocol allow you to automatically connect devices to the Bosch IoT Suite with minimal customization work.

Do you have particular interest in edge computing for your IoT project?
Have a look at the technical features of the Bosch IoT Edge, take a deep dive in our user documentation or subscribe to a free trial.

Your project requires both edge and cloud computing and you are looking for a custom tailored solution? Get in touch with our team in order to get started.