A white paper on data acquisition , analytics and cloud based on Edge Computing approach

Edge Computing in Industrial Application

Industry 4.0 Enabler

Sharma, Kawshol Devilal

IIoT, Industry 4.0, edge and cloud computing

The Industrial Internet of Things (IIoT), Industry 4.0 means different things to different people. End users from all sorts of industries and businesses incorporate IIoT elements to realize value in ways that suit their unique needs. In the most general terms, implementing an IIoT strategy involves connecting sensors and automated systems located in challenging manufacturing and process locations to create a unified data network.

This enables extensive remote data monitoring and remote data acquisition, autonomous machine-to-machine interaction, and deeper operational analysis, autonomous machine-to-machine interaction.

Common goals for users are smarter operations, improved equipment effectiveness, and cost reduction. But whether the user is the operations or engineering group in a plant, an original equipment manufacturer (OEM) who builds manufacturing machinery, or perhaps a systems integrator (SI) tasked with tying it all together, everyone agrees that IIoT implementation should be easy and secure. At the simplest level, IIoT implementation involves getting field data into cloud systems so it can be processed and shared among many users and applications.

In Industrial Context, this data is most often collected conventionally incorporating devices like programmable logic controllers (PLCs), human-machine interfaces (HMIs), and supervisory control and data acquisition (SCADA). While these systems can perform a certain amount of computing in the field or transmit raw data over to the information technology (IT) side of the business for additional processing, they have restrictions in ease of deployment, technology and options as compared with modern options available in the industry.

Traditional industrial communications architectures

Figure1 : Tradition IIoT communication architecture

As depicted above, Traditional architecture for the IIoT have demanded many layers of hardware and software, driving up cost and complexity and driving down performance. Hardware, software, and communication technologies are progressing and harmonizing to make IIoT implementations practical for any type of end user and application.

Traditionally, hardware had used proprietary protocols and media with a relatively limited scope of operation. This fact has led to problems associated with complex, layered system architectures, which were required to facilitate data processing and transmission. Typically requiring data conversion at multiple levels before the data can be shared across different systems.
Traditional models of sending all the data to cloud for higher level have made the cloud costs expensive. All analysis at cloud levels is sometimes inefficient for immediate actions as well. This has led to renewed interest and focus on edge computing.

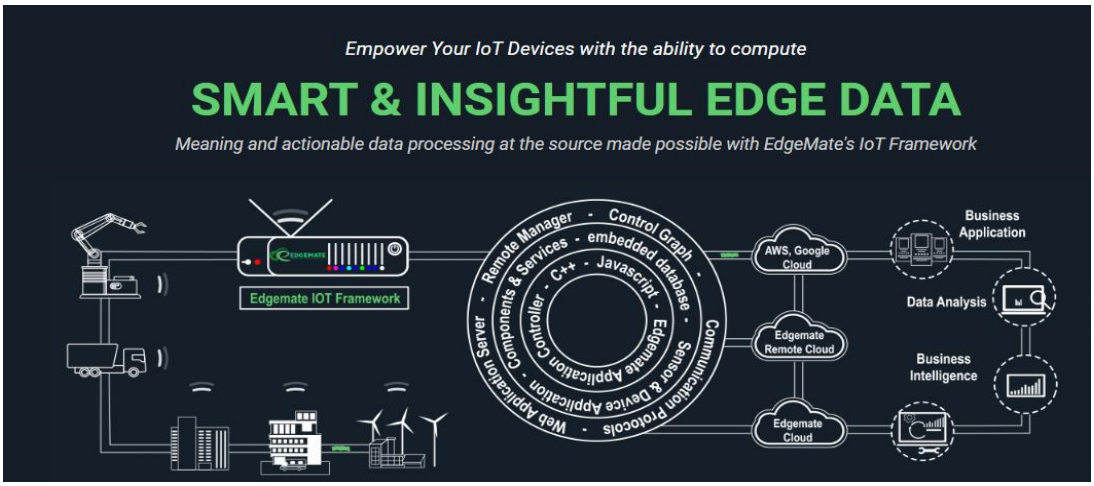
In the edge computing model, however, the data demands of large networks are served by embedding more computing power in the field, where data is produced. Rather than requiring a deep technology stack to move data from limited field devices to powerful central computing resources, edge computing devices are capable of processing data directly at the source and then transmitting it wherever it needs to go.

Industrial edge computing hardware addresses the complexity and security concerns of end users by flattening the communication architecture and introducing up-to-date IT standards to the OT domain. For instance, a traditional wired sensor can be connected directly to a field-installed, remote edge I/O device, which will establish an encrypted, certified connection directly to the cloud and transmit the signal data securely through its own internal firewall. This avoids the complexity of traditional systems, which would have required interposing PLCs, PCs, and security hardware.

This article discusses how new EdgeMate IIoT edge framework provide a simple and secure alternative approach to achieving different IIoT connectivity objectives. It presents the general architectural improvements that edge computing affords and explores their application in solving the specific challenges of the three user groups mentioned above: plant operations and engineering teams, OEMs, and SIs.

Solutions for Different Stake holders

While everyone is looking for solutions that are good, fast, and cheap, those three attributes rarely intersect. A better way to define the attributes of a robust and optimal IIoT solution is to examine the needs and goals of different end user groups.



Operational end users:

End users at manufacturing businesses and production plants all need good data. There are operators who rely on visualization to run their plants daily, process engineers who want to optimize operations, plant engineers who will expand systems when necessary, and maintenance teams who troubleshoot and correct issues.
EdgeMate implementation automatically delivers the right information to the right person at the right time.

OEMs create the machinery and equipment operated by end users.

They are experts in the equipment they build and automate, but once a machine ships it is not under their direct control on their own site; it could be anywhere in the world. To build better machines and improve support to existing clients, OEMs need IIoT solutions that enable remote connectivity and can gather performance data for troubleshooting and continuous improvement.
EdgeMate implementation will connect them with a highly distributed fleet of equipment across disparate networks and security schemes.

Systems integrators.

Of all the parties interested in using IIoT connectivity, SIs likely have the most automation specialists on board, with skill sets spanning platforms, technologies, and industries. Typically, however, they are also the service providers and subject-matter experts (SMEs) for a large customer base. They are tasked with the constant demand to win new business and provide long-term support for previous projects.

Edge Computing advantage

Traditionally, hardware has used proprietary protocols and media with a relatively limited scope of operation. This fact has led to problems associated with complex, layered system architectures, which were required to facilitate data processing and transmission.
Hardware, software, and communication technologies are progressing and harmonizing to make IIoT implementations practical for any type of end user and application. An important design technique underlying this evolution is called edge computing. It addresses some of the key challenges faced by end users by shifting the traditional communication hierarchy described previously toward a more distributed model.
In the edge computing model, however, the data demands of large networks are served by embedding more computing power in the field, where data is produced. Rather than requiring a deep technology stack to move data from limited field devices to powerful central computing resources, edge computing devices are capable of processing data directly at the source and then transmitting it wherever it needs to go.
Industrial edge computing hardware addresses the complexity and security concerns of end users by flattening the communication architecture and introducing up-to-date IT standards to the OT domain. For instance, a traditional wired sensor can be connected directly to a field-installed, remote edge I/O device, which will establish an encrypted, certified connection directly to the cloud and transmit the signal data securely through its own internal firewall. This avoids the complexity of traditional systems, which would have required interposing PLCs, PCs, and security hardware.

Technique underlying this evolution is called edge computing. It addresses some of the key challenges faced by end users by shifting the traditional communication hierarchy described previously toward a more distributed model.

Edge applications with EdgeMate

EdgeMate IIoT framework is based on https://macchina,io an OSGi, SoA based Service oriented “IoT Software development kit”.
Open Services Gateway Initiative (OSGi) is a consortium of more than 80 companies from around the world working together to create an infrastructure to enable the deployment of services over wide area networks to local networks and devices. (https://www.osgi.org/)
SOA, or service-oriented architecture, defines a way to make software components reusable and interoperable via service interfaces. Services use common interface standards and an architectural pattern so they can be rapidly incorporated into new applications. This removes tasks from the application developer who previously need to redevelop or duplicated existing functionality or had to know how to connect or provide interoperability with existing functions.(https://en.wikipedia.org/wiki/Service-oriented_architecture)
EdgeMate implementation will give them cost-effective and efficient ways to deliver systems to their customers, including proven security and multiple communication options for broad interoperability and remote access.
EdgeMate implementation satisfies the needs of all these parties will include, to varying degrees:

- Flexible scalability
- High interoperability
- Embedded security
- Good performance relative to cost
- Low administration and high maintainability
- Future implementation safe

The complexity and expense associated with the current traditional solution set oftentimes keeps end users from fully realizing these goals. In the next section we will examine some of the inherent limitations of the conventional approach and solutions provided by EdgeMate.

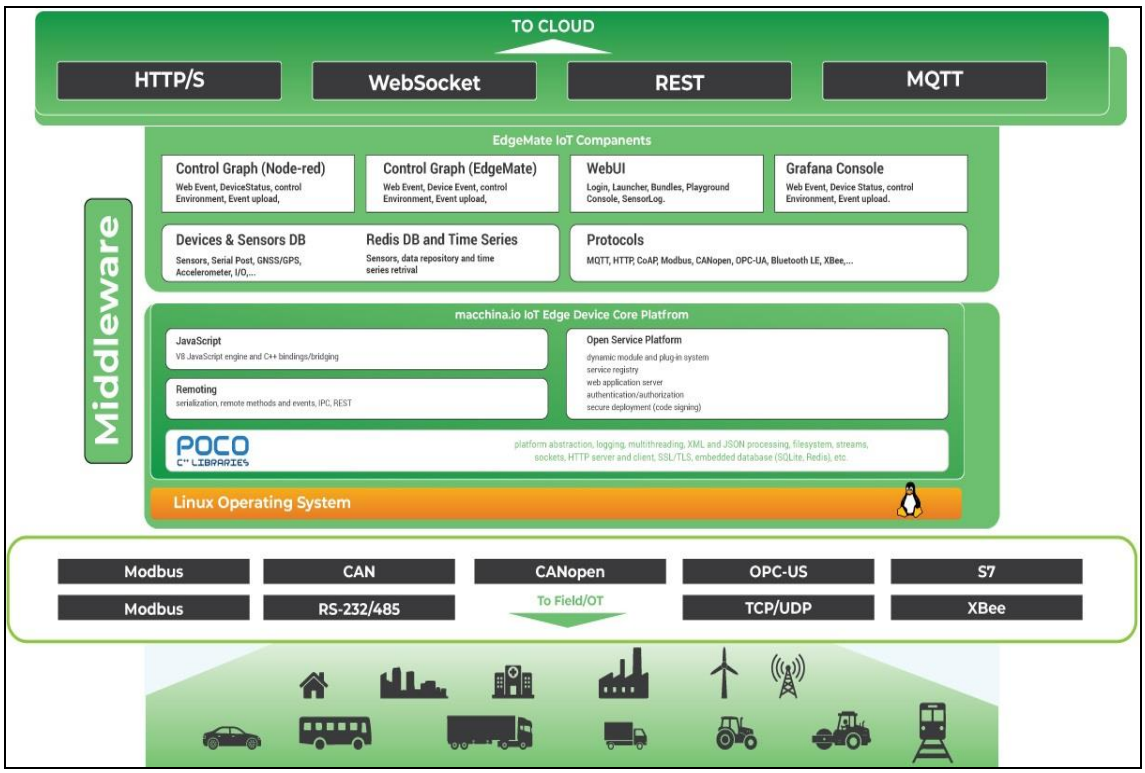


Figure2 : EdgeMate SW architecture

"EdgeMate Enabler" Advantage

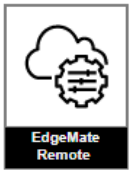
"EdgeMate Enabler" provides EdgeMate services and on the EdgeMate core application layer, API messaging so you can use to develop an efficient application w/o worrying of the underlaying complexity of OS and bus architecture bellow.

The various mechanisms can be freely combined for an optimum performance tailored to the specific characteristics of the devices and the communication networks.

They are briefly discussed in the remaining sections of this White Paper. Detailed information is available in the EdgeMate SW user manual specifying the service model and the Green Book specifying EdgeMate application layer messaging.

Enabler service	Description
Signal Database (SDB)   	Service applications want to interact with the IIoT agnostic to the bus or data collection architecture - SDB, through an abstraction interface allows access to sensors decoupled from how these signals are represented and encoded in the underlying bus. - SDB provides RAM data base with time-series data storage extension - SDB provides "Separation of data and metadata" - SDB provides "Aggregation of data" in RAM data base - SDB provides "Selective access" API to retrieve data in multiple ways - SDB provides "data bonds checker" to improve the reliability - SDB Avoids duplication of the code in each service.
Control graph 	Most applications want to process the data on the edge of network - Control graph tool allows the predefined operations to be organized in a sequence for data formatting, collection, or upload to remote. - Control graph allowed to store the data for use of other services in SDB, Provides a System aspects of the data collection pipeline - Applications of graph-based intelligence for control functions and operations - Graph can be of unsupervised or self-supervised depending on need - Graph can be updated new power cycle with changing of configuration files.
Remote Data collector service (DC)	- Automatically collect the data and upload - Remote configurable system - Data collection tool allows to create data collection scenarios - DC uses the internal Signal database (SDB), which makes the signals accessible for data collection - The synchronization mechanisms use "SignalGroups". The

	configurable “SignalGroup” defines transmission events and groups of individual signals available at SDB, those Signals contained in a “SignalGroups” could represent a service in backend. - The DC will keep controller awake depending on configuration of the “SignalGroups”. - DC triggers the upload of the collected data to the backend.
Power state manager 	- Power state manager service encapsulate Linux device power cycle - Help gateway to act according to power state - Provides a watchdog to the application - Acts as a supervisor for the process memory and CPU. - It provides power save modes for the battery driven operations
Node red workflow management 	Node-RED is an open-source programming tool, for connecting hardware devices, APIs and online services creatively and easily. - Supports browser-based flow editing making it user friendly, accessible, and visual - It is built on Node.js, which is a none-blocking, lightweight I/O model, making it lightweight and efficient - Flows created in Node-RED are stored using JSON, and can imported and exported and shared with ease - Node-RED can be run locally - Ability to run in cloud environments like Bluemix, MS-Azure, FRED etc - Node library is continuously growing - Built-in support for MQTT, Modbus, CoAP, AMQP, BLE, OPC UA, and even mDNS protocols
Web Console 	EdgeMate comes with A built-in web server providing different web apps for - IIoT device widget sample to display data - Device management apps for monitoring the device - Device software update interface - Testing and getting started with JavaScript development. These can be modified or replaced with a device-specific interface and apps. In any case, the web server is just another plug-in, so it can even be removed entirely if not needed.
Grafana console 	The Grafana project is a time-series and graph-focused dashboarding tool to make everything look cleaner and more elegant, with fewer things distracting you from the data. - Dashboard templating - Provisioning - Annotations - Kiosk mode and playlists: - Custom plugins - Alerting and alert hooks - Permissions and teams - SQL data sources: - Authentication
EdgeMate Remote Manger	- Devanture Remote Manager is edge tools for rapid device deployment and easier asset and inventory management it

	uses macchina remote manager stack - Monitor network, device and asset performance and security with bi-directional communications - Access data from edge devices that were previously out of reach and define precisely where you want the business logic - Automate mass firmware and software updates to stay in compliance - Integrate device data through open APIs to gain deeper insights - Receive detailed reports and real-time alerts on network health and device condition - Manage remote health of device
Security and Privacy  	- The REMOTE server only transparently forwards HTTP requests and TCP connections but does not store any data passed through it. - Both the connection between the device and the REMOTE server, as well as the connection between the client (web browser) and the REMOTE server are encrypted and secured with state-of-the-art TLS 1.2. This technology is inherently secure. - No open ports to the Internet, immune denial-of-service attacks against the device. - Devices authenticate themselves against the REMOTE server when setting up the tunnel connection, - Device authentication is done through a certificate based.

EdgeMate fully developed applications will look like figure3 ,

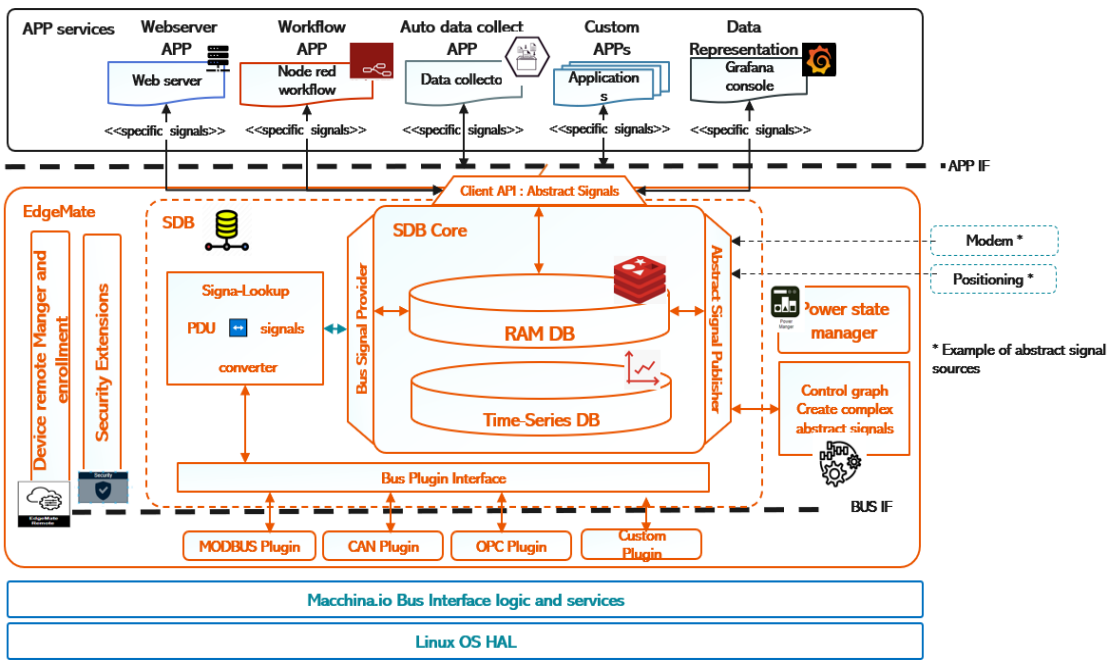


Figure3 : EdgeMate core services

Summary

As discussed in the previous sections, Edge Computing will become a formidable option for all Industry 4.0 and IIoT. Key points as listed below

- ✓ Edge Computing will have much simpler implementation layers as compared to Traditional Architecture proving to be much more cost effective, reliable, and convenient to implement.
- ✓ Edgemate provides a complete computing stack as required by OEMs, System Integrators as well as end users to achieve their application requirements
- ✓ Edgemate provides a suitable mix of different cloud layers merged for ease of deployment, cost optimization, secured connections for the user to access the data they need.