Core Technologies of Edge Intelligence for the IoT

This white paper introduces Advantech’s Edge Intelligence Server (EIS) that enables connectivity, data and device management plus analytics at the edge of the Internet of Things.

- An overview of the Internet of Things
- Core Technologies
- Software Offerings
- Conclusion

Author: Kurt Ou, Advantech Embedded-IoT Group Product Manager
# Table of Contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Introduction</td>
<td>2</td>
</tr>
<tr>
<td>Core Technologies</td>
<td>6</td>
</tr>
<tr>
<td>MQTT, RESTful API</td>
<td>7</td>
</tr>
<tr>
<td>Microservices</td>
<td>8</td>
</tr>
<tr>
<td>Docker Container</td>
<td>9</td>
</tr>
<tr>
<td>Software Offerings</td>
<td>10</td>
</tr>
<tr>
<td>EIS Software Architecture</td>
<td>11</td>
</tr>
<tr>
<td>Connectivity</td>
<td>12</td>
</tr>
<tr>
<td>Edge Intelligence</td>
<td>13</td>
</tr>
<tr>
<td>WISE-PaaS/RMM (WISE-Agent &amp;Node-RED)</td>
<td>14</td>
</tr>
<tr>
<td>Advantech add-on Nodes for Node-RED</td>
<td>15</td>
</tr>
<tr>
<td>Security</td>
<td>16</td>
</tr>
<tr>
<td>EIS RESTful API</td>
<td>17</td>
</tr>
<tr>
<td>Dashboard</td>
<td>18</td>
</tr>
<tr>
<td>Data Service</td>
<td>19</td>
</tr>
<tr>
<td>Application Services</td>
<td>20</td>
</tr>
<tr>
<td>WebAccess/SCADA</td>
<td>21</td>
</tr>
<tr>
<td>Development Tools</td>
<td>22</td>
</tr>
<tr>
<td>WISE-PaaS Marketplace</td>
<td>23</td>
</tr>
<tr>
<td>Conclusion</td>
<td>24</td>
</tr>
</tbody>
</table>
Introduction

This white paper introduces Advantech’s Edge Intelligence Server (EIS) architecture for the Internet of Things (IoT). The EIS includes software packages, core technology, IoT development tools, pre-configured WISE-PaaS software services, plus the flexibility to add more software modules from the WISE-PaaS marketplace. EIS helps customers build and launch innovative IoT applications and offer easy-integration solutions.

Obviously the IoT has enormous potential, but its evolution has been affected by product development challenges such as rapidly changing requirements and a wide variety of hardware and software technologies and applications. Customers encounter the following pain points in developing an IoT product:

- How to support heterogeneous sensors and actuators via the Internet
- How to integrate heterogeneous wire/wireless connectivity protocols (e.g., Modbus, LoRa, Sigfox, Wi-Fi, Bluetooth, etc.)
- How to easily port original software to different hardware (e.g., MCU, x86/ARM CPU, GPU, etc.) and OSs (e.g., Microsoft Windows, Linux Distributions, mbed OS, Android, etc.)
- How to easily connect to various cloud services (e.g., WISE-PaaS, Microsoft Azure, ARM mbed Cloud, IBM Bluemix, etc.)
- How to maintain data ownership and integrity and its implications for security and privacy
- How to deploy, update, upgrade, and maintain a large number of devices and services
- How to transform big data into valuable business information

Developing an IoT product usually requires solving problems concerning sensors, connectivity, security, cloud services, storage, device hardware, device maintenance, edge/cloud analytics, system integration, application development, and so on. One of the first challenges that many companies face is how to migrate to an IoT application while balancing design time, time-to-market, and risk.

IoT data can be large in terms of volume and applications typically have real-time requirements. Transmitting massive amounts of raw data over a network puts a load on network resources. Often it is more efficient to process data near its source and send only the valuable fraction over the network to a cloud center. Edge computing is a distributed information technology (IT) architecture in which client data is processed at the periphery of the network, as close to the originating source as possible. Time-sensitive data in edge computing may be processed at the point of origin by an intelligent device or sent to an intermediary server located in close geographical proximity to the client. Data
that is less time-sensitive can be sent to the cloud for historical analysis, big data analytics, and long-term storage.

Advantech’s EIS features edge computing architecture for its IoT solution. EIS enables local IoT networks to perform edge intelligence to maximize energy efficiency, reduce privacy threats, promote ease of implementation and modularization, and minimize latencies.

**Figure 1: WISE-PaaS IoT Software Platform Service**

As Figure 1 shows, Advantech WISE-PaaS IoT software platform services are based on three key components: the IoT Node, Edge Intelligence Server (EIS), and cloud service.

Advantech’s focus on its edge device development, is to ensure that the EIS software suite includes the southbound sensing device connectivity required to handle the diverse sensing protocols such as Modbus, OPC, BACnet, Wireless IP/NonIP, and so on. Those protocols are handled by plugin modules that take care of sensor data, data normalization, and communication, as shown in Figure 2.
The EIS then handles the northbound cloud connectivity and intelligence facilities through the use of the microservice container paradigm to modularize the different cloud connections and enabling device management. EIS intelligence facilities also adopt the microservice container architecture to support the data ingestion workload such as data pre-processing and cleaning. The most valuable faculty in the EIS software suite is the on-demand, real-time analytics service known as streaming analytics which extracts pre-set data features in real time as data is generated. The EIS Predictive Maintenance and Quality (PMQ) is provided as a proof of concept for edge field prediction, and the PMQ module can be updated via cloud service. The customer can leverage this framework to develop their own analytic module or PMQ module via the EIS open standard architecture. The entire EIS open standard is based on the ubiquitous MQTT communications protocol and the modularizing Docker container technology.

The edge interface offers a RESTful API, MQTT, and Node-RED to facilitate drag-and-drop application development. Node-RED and the configuration utility make it easy to implement a custom
applications through simple steps in a user-friendly UI. Moreover, the well-documented SDK with MQTT sample code and RESTful API interface allows the advanced developer to design and develop rich IoT applications that fulfill high-level requirements.

The last component is cloud services: Advantech provides a high-security service with SSL/TLS communications and leverages Intel® Security both on the edge device and for cloud security management. The data service can provide the PostgreSQL DB and Mongo NoSQL DB as standard offerings, and also supports a standard interface for integration with a wide range of data processing and storage products. The dashboard website serves as the IoT application user interface, and displays information via browser or mobile device through visualization facilities such as Azure Power BI or Tableau.

The WISE-PaaS IoT software platform provides a marketplace for sourcing diverse IoT software utilities, providing pure cloud solutions such as database, dashboard, and machine learning tools. Although most solutions on Microsoft Azure are 3rd party, Advantech offers additional IoT solutions such as cloud- and edge-ready packages for common requirements; these include the WISE-PaaS/RMM for Remote Monitoring and Management, WebAccess/SCADA for automation control plus more. Lastly, the marketplace has additional offerings in the form of analysis microservice containers such as HDD and key component PMQ for Predictive Maintenance and Quality, and more. So the customer can browse and purchase various types of IoT software solutions without having to deal with the complexities of development and installation. All of these facilitate customer IoT solution development, and accelerate edge intelligence, security, data and dashboard services.
Core Technologies

MQTT
MQTT is an extremely simple and lightweight publish/subscribe messaging protocol. It was designed for constrained devices and low-bandwidth, high-latency, or unreliable networks. The service publishes its capability and data to an MQTT broker and subscribes to specific topics for input interfaces.
http://mqtt.org/

RESTful API
A RESTful API defines a set of functions that developers can use to perform requests and receive responses via HTTP protocol, such as GET and POST. Because RESTful APIs use HTTP as a transport, they can be used by practically any programming language and are easy to test. It’s a requirement of a RESTful API that the client and server are loosely coupled and independent of each other, allowing either to be coded in any language and improved upon at will, which leads to system longevity and ease of evolution.

The RESTful API specifies what it can provide and how it can be used, and requires that details such as query parameters, response format, request limitations, public use/API keys, methods (GET/POST/PUT/DELETE), language support, callback usage, HTTPS support and resource representations should all be self-descriptive.

The RESTful architectural properties affected by the constraints of the RESTful architectural style are:
• Performance - component interactions can be the dominant factor in user-perceived performance and network efficiency
• Scalability to support large numbers of components and interactions among components
• The simplicity of a uniform interface
• Modifiability of components to meet changing needs (even while the application is running)
• Visibility of communication between components by service agents
• Portability of components by moving program code with the data
• Reliability - Resistance to failure at the system level in spite of failures of components, connectors, or data

https://www.sitepoint.com/developers-rest-api/
https://en.wikipedia.org/wiki/Representational_state_transfer
Microservice

EIS adopted the Microservice Architecture pattern as shown in Figure 3. Instead of building a single monstrous, monolithic application, the designer can split the application into a set of smaller, interconnected services. A service typically implements a set of distinct features or functionalities, such as connectivity management, vertical application, etc. Each microservice is a mini-application that has its own architecture consisting of its business logic along with various adapters.

Benefits:
1. Individual services are much faster to develop, and much easier to understand and maintain
2. Each service may be developed independently by a team that is focused on that service
3. Each microservice can be deployed independently, with continuous deployment possible
4. Each service can be scaled independently

https://www.nginx.com/blog/introduction-to-microservices/

Figure 3: Microservice Architecture
Docker Container

Containerization – is an OS-level virtualization method for deploying and running distributed applications without launching an entire Virtual Machine (VM) for each application. Instead, multiple isolated subsystems, called containers, run on a single control host and access a single kernel. As shown in Figure 4, containers share the same OS kernel as the host; containers are usually more efficient than VMs, each of which requires a separate OS instance.

A Docker container wraps up a piece of software in an independent subsystem, complete with file system and everything it needs to run: code, runtime, system tools, and system libraries – anything that may be installed on a server. This guarantees that it will always run the same, regardless of the environment in which it is running.

Containers hold the components necessary to run the desired software, such as files, environment variables, and libraries. The host OS also constrains the container's access to physical resources – such as CPU and memory – so a single container cannot misbehave and consume all of a host's physical resources.

Figure 4: Docker Container Architecture
Software Offerings

Advantech EIS is a complete, scalable, flexible hardware and software IoT Edge Intelligence Suite. EIS can be customized, combining several software services; it is then installed on different hardware depending on requirements. Advantech Edge Intelligence Suite includes:

- EIS Microservice Architecture
- Connectivity (e.g., MQTT, Wi-Fi, Bluetooth, Modbus/Modbus TCP, and so on)
- WISE-PaaS/RMM, WISE-Agent, Node-RED
- Advantech add-on nodes for Node-RED
- Security for device, data, and communications channel
- EIS RESTful API
- Dashboard – Freeboard on RMM and Node-RED-UI on EIS
- Data Service
- WebAccess/SCADA
- A set of application development tools
- WISE-PaaS Marketplace

EIS Software Architecture

![EIS Software Architecture Diagram]

Figure 5: EIS Software Architecture
Figure 5 shows the EIS software architecture, and illustrates the EIS design concept. (Note that not all depicted services are in production at the time of this writing and the customer should check with Advantech for specific component availability.)

The EIS Software Architecture is classified into 5 category layers. Each function is implemented as its own microservice, using the MQTT broker as the communication bus. All microservices interface with other microservices or clients. At runtime, each instance is a Docker container. This makes it easy to deploy distinct experiences for specific users, devices, or special-use cases. The customer can easily develop and deploy services using EIS SW architecture.

- The first or bottom layer of the architecture is the wired and wireless sensor network connectivity layer. Wired sensors support various types (e.g., SCADA, Modbus, OPC-UA, etc.). Wireless connectivity can also be of various types (e.g., Wi-Fi, Lora, etc.). The network connectivity layer is charged with collecting data, and managing sensor hubs, translating sensor protocols to the MQTT protocol, and then passing data to the MQTT communication bus.
- Second, the SDK Layer provides various software services such as EIS RESTful API, HDD Fault Prediction Algorithm Service, and so on. The developer can call these services through the RESTful API or the MQTT protocol. The user can add its own service in this layer (e.g., Machine Learning Platform, Data Base engine, and so on).
- Third, the Flow-based Layer provides Node-RED as the data flow design engine. The customer can easily wire together flows using Advantech add-on nodes such as SUSI API, WSN, and HDD prediction nodes. This helps customers quickly and easily design logic paths via simple drag-and-drop operations on a graphical environment.
- Fourth is the Management and Presentation UI Interface Layer. EIS provides a Webmin for system administration and IoT connection configuration using the Node-RED-UI for presenting IoT/sensor data.
- Fifth is the Cloud Layer. EIS is pre-installed, with the WISE-Agent connected to WISE-PaaS/RMM Cloud Server. Advantech also provides 3rd-party Cloud Service Agents (e.g., Microsoft Azure, IBM Bluemix, ARM mbed Cloud, etc.).

**Connectivity**

EIS uses the MQTT broker as the asynchronous communication bus for inter-service communication. The inter-service communication protocol is based on the MQTT standard and the IETF (the Internet Engineering Task Force) defined media types for Sensor Markup Language (SenML) in JSON format. The user can follow EIS communication protocol or use the “WISE Sensor Network Abstract Interactive Layer” (a lightweight WISE-Agent) SDK to integrate a new service within the EIS eco-system.
**Edge Intelligence**

- **WISE-PaaS/RMM**

  The WISE-PaaS/RMM (Remote Management and Monitoring) is a combination of two major components. A server-side system (the Device Manager, or DM) communicates with the devices via various protocols and provides both individual and bulk device control. It also remotely manages the software and applications deployed on the devices. It can monitor and/or reset device applications or processes if necessary. The Device Manager works in conjunction with the device management agent (WISE-Agent). There are multiple different agents for different platforms and device types. The Device Manager also maintains a list of device identities and maps these identities to owners. It also works with the Identity and Account Management block to manage device access controls (e.g., Who else can manage a device apart from the owner? How much control does the owner have vs. the administrator? etc.).

**WISE-Agent**

![Figure 6: WISE-Agent Architecture](image-url)
WISE-Agent is a software development framework for communications between devices and the WISE-PaaS/RMM Server. The WISE-Agent framework as presented in Figure 6 provides a rich set of user-friendly, intelligent and integrated interfaces; this speeds development, enhances security, and makes it easy for agent applications to communicate with the RMM Server. The framework has three benefits:

- **Standardization** - The MQTT protocol forms the basis for communicating and exchanging data with the WISE-PaaS/RMM Server. The IoT sensor data report format follows the SenML in JSON format.
- **Portability** – The whole framework is written in C language and follows the ANSI C Standard. A C compiler is available for most systems, and it is often the first compiler provided for a new system.
- **Scalability** - The WISE Agent Framework is based on functional partitioning into discrete, scalable, reusable modules, and supports plug-and-play.

**Node-RED**

Node-RED is available as open source, and is implemented by the IBM Emerging Technology organization. It provides a browser-based flow editor that easily wires flows together using the wide range of nodes in the palette. Flows can then be deployed to runtime with a single click. The flows created in Node-RED are stored using JSON, and can easily be imported and exported for sharing with others. With built-in node.js, it can be run at the edge of the network or in the cloud. The node package manager (npm) ecosystem is easily used to extend the palette of nodes available, enabling connections to new devices and services.
Advantech Add-on Nodes for Node-RED

SUSI Nodes
A collection of Node-RED nodes access Advantech platform hardware functions through node-susiiot and node-susi.

UI-Flow-Creator Nodes
Nodes collocate with SUSIIoT APIs and node-red-dashboard nodes to create a customized dashboard. The ui-creator node uses SUSIIoT to get capability and device status from platforms, and automatically constructs a Node-RED flow to represent the customized dashboard. In addition to displaying device status, the created dashboard provides a device control mechanism. User input values on the dashboard are redirected to the input of a ui-creator-ctrl node, then the ui-creator-ctrl node sends the values to the device through SUSIIoT APIs.

WISE-PaaS/RMM Nodes
Node-RED nodes provide for communications between Node-RED and WISE-Agent. A WISE-PaaS/RMM Sender node sends data to the WISE-Agent; the WISE-PaaS/RMM Receiver node receives messages from the RMM agent; and WISE-PaaS/RMM Responder node responds to RMM RESTful requests.

WISE-PaaS/RMM nodes can collocate with existing Node-RED nodes; an edge device can easily pass its data to an RMM server, or it can receive control commands from an RMM server without developing RMM agent plugins for different protocols/devices. For example, if data needs to be read from a Modbus TCP device and then passed to an RMM server for later use, a Modbus TCP node (provided by a 3rd party) can be wired together with a WISE-PaaS/RMM Sender node.

WSN Nodes
Wireless Sensor Network (WSN) is supported on Advantech IoT Gateway and all provided features are basically described by a RESTful API. To give users quick access to WSN features so they can build and prototype application control flows without dealing with RESTful API manipulations and programming, Advantech provides the following Node-RED nodes, which make it easy to read information and access WSN functions on EIS.

Security
Security is the key to establishing trust in the IoT. Edge Intelligence Server (EIS) offers IoT application builders the most effective way to secure critical infrastructure and devices against
internet-based threats. For communication, an SSL certificate is used to encrypt communication between Server and Client sides. EIS also implements TLS and DTLS for encrypted data transmission. For operating systems, Intel® McAfee secures embedded devices and automates the enforcement of software change-control policies, helping manufacturers ensure that their products and devices are protected from cyber threats and attacks. McAfee solutions span a wide range of technologies to address the above challenges, including application whitelisting, change management, and integrity.

- **EIS RESTful API**

![EIS RESTful API](image)

*Figure 8: EIS RESTful API*

The EIS RESTful API is the entry point for all EIS software services. The RESTful API handles requests in one of two ways; they are either simply routed to the appropriate service, or fanned out to multiple services.

RESTful API service gathers pre-process data and generates REST APIs for EIS software service as seen in Figure 8. The RESTful API is also responsible for tasks such as load balancing, caching, access control, API metering and monitoring, and can be implemented effectively using NodeJS express. The customer can easily and quickly develop an application, designing its data flow and web application using the RESTful APIs.

- **Dashboard**

EIS provides the Dashboard tools that make it easy to decipher what data is saying, so management can make analytics-based decisions to steer their businesses. It enables the IoT solution provider or end user to create automated, repeatable, and management-oriented summaries of IoT
data. The dashboard includes Freeboard in the RMM Cloud Server, and Node-RED-UI in the EIS system.

**Freeboard**

Freeboard provides simple, real-time visualization of key performance indicators. This easy tool opens up many possibilities for IoT projects because it’s simple, affordable, open source, and ready for extension. Customers can get started for free and then when it is time to ramp up, they can select a plan that’s right for them.

Freeboard is built upon a secure, high-performance, enterprise-class cloud system; it offers seamless integration with dweet.io, with access to any web-based API. Customers can select from a growing list of included widgets, or add their own. They can design layouts that meet their exact needs, and modify them easily when requirements change. Some keep their Freeboards public and pay $0. Others select one of the low-cost plans to make them private. Any Freeboard can be duplicated and used as a starting point for a new one (permission required). Every Freeboard has a unique URL that can be shared via email, SMS, and social networks.

**Data Service**

EIS is on the cutting edge of computing applications, managing data and services between sensors, network, and cloud server. Some edge intelligence devices may connect only intermittently such as laptops, smartphones, and tablets, but data is usually generated more or less continuously and requires continuous management.

Advantech’s EIS provides these types of data services: PMQ, data at rest, and data in motion. The term PMQ was recently coined by IBM, and it’s a software solution that leverages the power of predictive analytics to anticipate and prevent critical breakdowns using out-of-the-box data models that include Machine Learning. Any asset-intensive businesses including energy, medical, transportation, automotive, and manufacturing can benefit from this. Predictions and alerts for HDDs and SSDs are implemented in the Docker container and use Machine Learning models.

Data at rest, or static data, is stored in a digital form. It is then used in EIS, even when network connection is absent. Data in motion is also used in EIS. The collection process for data in motion is similar to that for data at rest, but with different analytics. For data in motion, the analytics occur in real-time, either continuously as sensor data pours in, or when specified events happen. The system analyzes and filters sensor data before taking further action. Event triggering provides automated responses that deal directly with time-sensitive issues. Data at rest and data in motion are implemented in the nodes of Node-RED.
EIS offers many advantages by taking on the data service role:

1. It significantly decreases the data volume that must be moved, the consequent traffic, and the distance the data must go, thereby reducing transmission costs, shrinking latency, and improving QoS.

2. It’s not at the core of the computing environment, but it does limit and remove a major bottleneck and a potential point of failure.

3. Security is also improved as encrypted data moves from the extremities into the network core. The data is checked as it passes through protected firewalls and other security checkpoints where viruses, compromised data, and active hackers can be detected early on.

4. The ability to "virtualize" extends scalability. EIS virtualizes using Docker container technology and could use Docker Swarm to extend scalability.

**Application Services**

- **WebAccess/SCADA**

  Advantech WebAccess is a browser-based software package for human-machine interfaces (HMIs), and supervisory control and data acquisition SCADA. It is used to automate complex industrial processes for situations where remote operations are needed. All the features found in conventional HMI and SCADA software packages are available through an ordinary browser including Animated Graphics Displays, Real-time Data Control, Trends, Alarms, and Logs.

  WebAccess supports ample drivers including Advantech I/O modules, controllers and major PLCs, and standard protocols such as ModBus, OPC UA, OPC DA, and BACnet protocols. It is easy to integrate 3rd party software such as MES and ERP via open-interface WebAccess APIs.

  Previously, WebAccess could only monitor the status of sensors and devices. Now with the integration of WISE-PaaS/RMM, large amounts of sensor and device data can be uploaded to the WISEPaaS/RMM cloud to generate data statistics and analysis diagrams from the WISE-PaaS/RMM Dashboard, as well as to monitor equipment and platform statuses such as CPU temperatures, usage, board temperatures and so on in Remote Equipment Monitoring.
Development Tools

EIS also provides a set of development tools to help customers easily and quickly implement IoT applications in the EIS system. EIS development tools include:

- Node-RED, for flow-based program engine
- Advantech add-on nodes and sample flows for Node-RED
- Development SOP
- WISE-Agent plug-in mode
- IoT Hub development toolkit and samples

More Software Solutions on WISE-PaaS Marketplace

Device Management Package

Advantech WISE-PaaS Marketplace provides a Device Management solution package for an EIS system. The Device Management package includes WISE-PaaS/RMM Remote Monitoring and Management Software plus a Microsoft Azure Virtual Machine, which helps users build cloud-based management platforms easily and quickly.

WISE-PaaS/RMM is focused on IoT device remote monitoring and management, bridging layers of IoT Platform Architecture, and anchoring other domain-specific cloud applications. Azure Virtual Machines let the customer create and use virtual machines in the cloud. Providing what’s known as Infrastructure as a Service (IaaS), virtual machine technology can be used in a variety of ways.
The Azure Virtual Machine (VM) is one of several types of on-demand, scalable computing resources that Azure offers. Typically, VMs are chosen because they offer more control over the computing environment than other options.

An Azure VM gives customers the flexibility of virtualization without having to buy and maintain the physical hardware that runs it. However, customers still need to maintain the VM, taking care of configuration, patching, and installing the software that runs on it. Azure virtual machines can be used in various ways. Some examples are:

- **Development and testing** – Azure VMs offer a quick and easy way to create a computer with the configurations required to code and test an application.

- **Applications in the cloud** – Since demand for an application can fluctuate, it often makes economic sense to run it on a VM in Azure. Capacity, and cost, can be adjusted up or down by adding or subtracting VMs as needed.

- **Extended datacenter** – Virtual machines in an Azure virtual network can easily be connected to any organization’s network.

The number of VMs that a particular application uses can scale up and out to whatever is required.
Conclusion

The scope of the Internet of Things is evolving rapidly along with associated technologies. Advantech’s EIS architecture is based on practical, real world projects that have been deployed with customers to support a wide range of IoT capabilities. Advantech’s IoT architecture is useful, scalable, robust, deployable, and effective.

Advantech is experienced in a wide range of IoT solutions that provide connectivity for sensors and actuators on the Internet. Advantech also partners with well-known companies (e.g., Microsoft, ARM, ST, TI, IBM, etc.) to provide numerous kinds of hardware/software and cloud solutions.

Advantech Edge Intelligence Server (EIS) is a flexible and scalable hardware/software architecture that helps companies develop complex IoT infrastructure in an integrated ecosystem that serves different vertical markets. EIS helps bridge the technology gap and simplifies IoT deployment.

For more information, please visit the following websites:

• EIS Solution: eis.advantech.com
• WISE-PaaS IoT Software Platform: http://wise-paas.advantech.com/embedded
• WISE-PaaS Marketplace: http://wise-paas.advantech.com/marketplace