Common Industrial Cloud Interface – Uses Cases and Technical Requirements for Data Transfer

Stephen C. Briant
Technology Manager
Rockwell Automation

Thomas Whitehill
Remote Services Architect
Schneider Electric

Presented at the ODVA
2017 Industry Conference & 18th Annual Meeting
February 21-23, 2017
Palm Harbor, Florida, USA

Abstract

The Industrial Internet of Things (IIoT) is here, bringing new technologies, challenges and opportunities to industrial automation. Companies are looking to the internet and cloud computing to provide new ways to improve operations, increasing productivity and generate more revenue. Acquiring data from devices is a primary focus of IIoT in the market today, but there are definitely opportunities to do more. What are the challenges presented by these relatively new applications? What technologies are being leveraged to solve these challenges? What capabilities are needed to allow CIP Devices to provide an advantage in these applications? This paper will cover various applications or use cases and discuss the challenges presented by these applications. Several open communication protocols in use today will be examined as possible solutions to these challenges. Opportunities to leverage or extend existing standards, including ODVA standards, will be discussed.

Keywords

Device, Cloud, Gateway, Interfaces, Messaging, Storage, Analytics, Streaming, Big Data, IIoT

Introduction

The Industrial Internet of Things is here, bringing new technologies, challenges and opportunities to industrial automation. Companies are looking to the internet and cloud computing to provide new ways to improve operations and increase productivity. Acquiring data from devices is a primary focus in the market today, but there are definitely opportunities to do more. It is with these thoughts in mind that ODVA announced the formation of a new Special Interest Group (SIG) for the Common Industrial Cloud Interface, (CICI).

This new SIG, the latest in a group of ODVA activities under the Optimization 4.0 banner, will develop standards that enable new cloud applications to be developed by the member community leveraging the rich data available in devices that conform to ODVA standards. The new SIG intends to leverage “cloud technologies” available today and “connect” them with the rich information defined in CIP Devices in a simple and secure manner. The SIG will focus on making available means to discover CIP devices and the data available in those devices. The SIG will also look for opportunities to manage devices and collections of devices through gateways.

Cloud computing offers many advantages that were previously unavailable, starting with the ability to connect to devices across an enterprise or a machine type across multiple enterprises. In addition, the ability to scale computing power and storage are enabling new possibilities for analyzing data streams. In the sections below, we will highlight some of the basics of cloud computing.

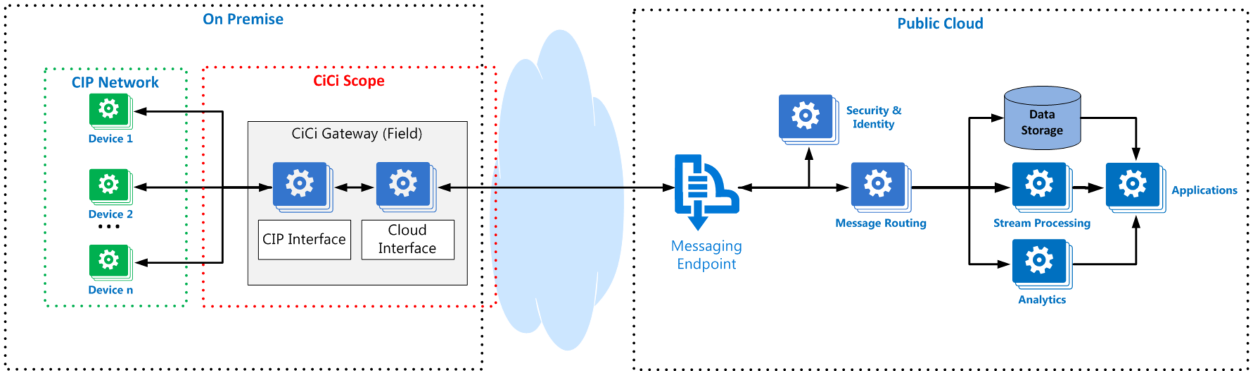
While many of the technologies available to construct cloud applications are familiar, there are also many new technologies that are being used today. We will cover some of those technologies that are relevant to connecting, discovering and moving data to the cloud. In addition to covering these technologies, we will discuss some guiding principles that will be followed when defining a standard and the resulting applications that will be enabled by the standard.

Next, we will explore a set of Information exchange patterns that are common to all applications that include devices, gateways and cloud. These communication patterns will form a basis for describing applications or use cases that the SIG intends to address.

Finally, a simple Proof-of-Concept that leverages open standards to communicate with Microsoft’s Azure cloud services will be provided. This will tie together the sections of the whitepaper while applying one of the information exchange patterns.

Reference Architecture

Reference Architecture



Definitions

The three main architectural components in the preceding reference architecture are CIP Device, CICI Gateway and the Cloud as defined below.

CIP Device

In the context of CICI reference architecture, a Device is a CIP-enabled device connected to a CIP-enabled network. The CIP Device's purpose doesn't change within the CICI context, nor do its characteristics or behaviors. This statement is very important to consider when contemplating the role of the CICI Gateway and Cloud as well as the Guiding Principles spelled out later in the document.

Only conceptually does the CIP-enabled Device change. In the CICI Reference Architecture, a CIP Device engages in an extended, much wider architectural context that includes the Cloud. In this wider architectural context, a CIP Device's data (telemetry, etc.) can be published (indirectly) to the Cloud, to be processed, stored and analyzed by services and applications executing there. Likewise, commands and notifications can originate from those same cloud-based applications and be pushed back down through the CICI Reference Architecture to CIP Devices. This is made possible because of the CICI Gateway (see below).

No changes to a CIP Device are required for it to interoperate within the CICI Reference Architecture. CIP Devices in CICI reference architecture are decoupled from the Cloud – all integration requirements are handled by the CICI Gateway. This important requirement enables full CICI interoperability for legacy CIP Devices.

CICI Gateway

A CICI Gateway is a middle-tier architectural component that is physically located on-premise and that logically bridges between the CIP network and the Internet and Cloud. Logically, it has two logical interfaces: downstream connects to the CIP-enabled network and n-number of CIP Devices; upstream connects to the Internet and Cloud.

As a middle tier CICI Gateway maintains contextual information for n-number of CIP Devices on its downstream interface. On its upstream interface, it maintains context (e.g. security credentials, endpoint URIs, etc) of multiple Cloud-based services, messaging systems.

A CICI Gateway performs a number of roles, most visibly bi-directional secure routing of Device-to-Cloud (D2C) and Cloud-to-Device (C2D) messages between CIP Devices on its downstream interface and Cloud-based services and applications on its upstream interface.

A CICI Gateway translates and normalizes message payloads as they pass across the domain boundaries of CIP and the Cloud. This crucial operation satisfies a core Guiding Principle, CIP Stays Home.

Logically, a CICI Gateway can be implemented at the level of a CIP Device, e.g. a CIP Device performs its own CICI Gateway functions. This approach is a two-tier, device direct to cloud connectivity pattern and is not recommended for reasons described in the Guiding Principles section.

Technically, a CICI Gateway is presumed to have adequate compute resources to service n-number of downstream devices, routing rules, translation rules and network operations. It is possible, therefore, to envision a Gateways which performs local, on-premise analytics and operational control 'at the edge'. However, these extra-capabilities are out of CICI scope at this time.

Cloud

In the CICI reference architecture, the Cloud is the Public Cloud. Technically, the public cloud (as describe elsewhere) is a complex collection of cost effective, scalable, geographically distributed infrastructure, software and platform services. Conceptually, however, the Cloud should be thought of in simpler terms: as the data collection, processing and analytics platform on which value will be created for the next generation of CIP customers.

Value for CIP customers is provided by 'actionable information' which can be used to make asset management and business decisions. Actionable information is derived by the CIP Device related data. Once collected, processed and analyzed, this data results in the ultimate value for ODVA customers: actionable asset information. Cloud-based applications will also consume raw CIP Device data as well as the derived actionable data.

Deriving actionable data is the ultimate goal of CICI since this is generally where value is provided to customers. For example, high levels of value to customers due to cost reduction can be achieved in the area of asset maintenance. As device monitoring is increasingly understood and analyzed, maintenance action can evolve from costly run-to-failure (reactive), to more efficient preventive (proactive) to the most cost effective, predictive.

Introduction to Cloud

This section introduces 'Cloud Computing' concepts as related to CICI so that readers can share a basic understanding of this complex technical landscape. The concepts discussed here are Public Cloud Computing and its core 'Service' models - Infrastructure-as-a-Service, Platform-as-a-Service and Software-as-a-Service.

Public Cloud Computing

Public Cloud Computing, aka 'the Cloud' or 'Cloud Computing', represents a significant, disruptive shift from traditional information technology (IT) and software product creation and delivery. In some cases, the Cloud will compete directly with an existing technological model, such as with on-premise hosted Data Center, potentially displacing it. In other cases, the Cloud should be seen, not as a competitive or displacing force to existing technologies, such as CIP, but as a complementary extension and enabler for developing new and different types of applications and solutions which leverage existing technologies.

Irrespective of it being a competitive or complementary force, Cloud Computing is disruptive to both technology and business models and brings significant risk for companies and industries attempting to adapt. The simple reason for this is that modeling and implementing Cloud-based solutions such as IIoT is complicated and multi-dimensional – this is not just a question of technology, but extends to business models, the increasing emphasis of OPEX over CAPEX, difficulties with costing imprecision, and the reformation of sales channel and revenue stream, etc. Understanding and making sense of these complications is a difficult challenge.

The first big challenge is that the Cloud Computing marketplace evolves daily and presents a bewildering and ever expanding set of choices among technologies, service offerings, service costing models and vendors. Secondly, the landscape of public Cloud vendors is complex and confusing: small, medium and large vendors compete for business in general purpose and specialized niche service markets. Finally, because of the rapid evolution of the Cloud there is no single path for successful Cloud adoption for a company or even its individual lines-of-business.

However, for companies and industries that are able to navigate these complications and can adapt to and leverage the Cloud's capability new possibilities emerge for developing customer value and driving business opportunity.

Cloud Computing offers companies extensive infrastructure, platforms and software services that traditionally have been available only at the corporate, enterprise or data center infrastructural level. These service groupings are referred to as 'as-a-Service' resources and are as follows:

- Infrastructure-as-a-Service (IaaS) is a self-service model for using storage, servers, networks and related virtual resources. The user of the service is usually responsible for managing applications, data, configurations, licensing and updates of related resources, including in some situations, the operating system. Resources usually have a number of options for performance and quality of service levels and can be very cost effective. A common use of IaaS is for hosting, networking, load balancing and storage for applications.

- Platform-as-a-Service (PaaS) is vendor managed platforms and middleware on which vendor applications and solutions can be rapidly developed, tested, operated and maintained. Messaging middleware, integration frameworks, databases and business process management are common PaaS services. These services are rapidly leveraged and require no maintenance. Similarly, to IaaS services, PaaS services offer a choice of service level and/or are billed per a consumption-based subscription model. Common uses of PaaS are the deployment of non-monolithic services, integrating decoupled components in a distributed system, storing of quantities of information at Big Data scale and deploying analytics algorithms.

- Software-as-a-Service (SaaS) is software deployed in the Cloud yet accessible virtually anywhere by clients. SaaS represents the large and growing field of cloud-based application services and offers as well as new architectural concepts such as microservices. SaaS applications are often accessed via thin-clients such as Web Browsers and mobile applications (native or hybrid) and, therefore, generally require very limited downloaded installation components or none at all. A common use (and ultimate goal) of SaaS for vendors is to use it as a platform for deploying products and solutions which include capabilities that are simply not possible with traditional on-premise deployment: Big Data analytics; geographic redundancy; IoT-level scalability; and world-wide access.

Introduction to Cloud technologies

Cloud computing has introduced, or at least raised to a high degree of public awareness, many new technologies and concepts. A sampling of these technologies are described below:

Big Data[1]

Big Data describes data sets of magnitudes and complexities never before seen. Big Data implies analysis of these data sets using specialized tools, algorithms programming languages and technologies to discover trends and patterns which may be used to drive business value. Data sets are characterized by the 'Four Vs', described below.

Four Vs of Big Data[2]

The Four V's of Big Data are Volume, Velocity, Variety and Veracity.

- **Volume**: Represents the scale of the data sets. One hears of systems running at 'Big Data Scale', which implies that they can consume and process data sets of unimaginable size. Technologies been built exclusively to address the need to scale with the many orders of magnitude expansion of storage needs. An example of this is NoSQL databases, which can scale to Big Data scale in IoT systems, replacing SQL databases which cannot.

- **Velocity:** Represents the ubiquitous and persistent collection and delivery of data, from consumers, sensors, systems into cloud-based Big Data oriented-systems

- **Variety:** Represents the ability to combine dissimilar data sets which previously could not have been combined are now being processed together. The Cloud enables the combination of public and private data to provide insights previously unattainable. Tweets, traffic, weather, Facebook, stock prices postings are examples of well-known dissimilar data sets which could be combined and analyzed to provide previously unattainable analytic outcomes. Industrial examples of public/private data sets are current power costs, impending weather events, raw materials cost, motor/machine/plant efficiency metrics, etc.

- **Veracity:** Represents the fact that much of Big Data information available is of questionable quality. Inaccuracy of data in consumer-based Big Data applications is a challenge, requiring cleansing and validation processing and assessment. It may not pose such a challenge in more tightly managed and audited Industrial environments. Nevertheless, it must be accounted for.

NoSQL[3]

NoSQL is a database model/approach which has evolved during the Big Data era and which contrasts markedly with Relational Database Management Systems (RDBMS or SQL): NoSQL databases have relaxed or non-existent referential integrity requirements, flexible storage options and limitless scalability and redundancy built in from the ground up. The acronym NoSQL originated from 'non SQL' and/or arguably 'not only SQL'.

[1] https://en.wikipedia.org/wiki/Big_data

[2] http://www.ibmbigdatahub.com/infographic/four-vs-big-data

[3] https://en.wikipedia.org/wiki/NoSQL

AMQP[4]

AMQP, Advanced Message Queuing Protocol, is the open standard and has emerged as a very popular protocol for sending messages to and receiving messages from Cloud-based systems. In addition to being open and standard, AMQP was designed with these characteristics Security, Reliability, Interoperability.

MQTT[5]

MQTT, Message Queue Telemetry Transport, is an ISO standard (ISO/IEC PRF 200922), publish/subscribe, lightweight messaging protocol used for Cloud-connectivity for limited network bandwidth and remote applications.

JSON[6]

JSON, JavaScript Object Notation, is a terse, readable, structured data format. It is very popular as a payload format for Device-to-Cloud and Cloud-to-Device messaging. A benefit to using JSON is that many stream processing applications are built to natively consume JSON structures efficiently and cost-effectively. Below is a very basic JSON message:

```
{
        “name”=”CICI”,
        “message”=”Hello World!”
}
```

Guiding Principles (or Concerns)

As we look to apply cloud technology and cloud communication patterns to uses cases, a number of concerns have been identified. These concerns form the basis for a set of “Guiding Principles” that will be applied to any resulting work of the SIG.

- Fundamental differences in Cloud-based vs CIP-network based application development

  The technical, semantic and application differences between cloud-based and CIP Device-based ecosystems is vast. The purpose of CICI is to define an integration approach to bridge between the two environments so that cloud-based applications (and application developers) can consume CIP Device data and produce actionable information useful to the CIP Device, directly or indirectly. It is not the purpose of CICI to replicate the CIP network-based ecosystem in the Cloud but to abstract and represent CIP Device data so that its use by Cloud applications is simple and efficient. The following are some of fundamental differences of Cloud-based from CIP-network based application development:

  - Distribution: Cloud-based applications are intrinsically distributed, combining resources from multiple compute, storage and service platforms to achieve function.

  - Real-Time: due to its distributed nature and platform dependencies, the notion of real-time is an uncommon concept in public Cloud computing.

[4] https://www.amqp.org/about/what

[5] http://mqtt.org/

[6] http://www.json.org/

    - Protocols and Payloads: Cloud-based application developers expect message payloads that are easily programmatically digestible, potentially extensible for business process purposes and normalized to contain meaning for a given application domain, such as Asset Management analytics. Abstracting CIP details satisfies these expectations while exposing CIP protocol or payload formats would cause lost productivity for developers, potentially large, incremental, profit impacting unmarshalling/marshalling costs.

    - Applications: Cloud-based applications have very broad scope and virtually no limitation to the types of functions that can be performed. Cloud-based applications can be modified and updated very rapidly (in minutes, hours) and often integrate multiple streams of information simultaneously to produce results

    - Communications between entities on the cloud are generally based on widely used open standards which are not industry specific and which may be replaced at any time. CIP communications, while standardized, are industry specific and slow to change.

    - Developer Pool: Cloud developers master a variety of programming languages, 'cloud oriented development' and cloud architectures that are very different from those needed by CIP network application developers. Abstracting CIP details with a normalized Device Asset domain model, for example, (see CICI Gateway) greatly increases the pool of potential developers for CICI-based Cloud applications

- In order to not compromise security of on-premise resources, CICI Gateways and CIP Devices are not exposed the internet by publicly available interfaces such as publish/subscribe broker or http web service endpoints. Device-to-Cloud and Cloud-to-Device communications are through an established CICI Gateway. For example, to support Cloud-to-Device communications, Cloud applications send messages to devices indirectly through Cloud-based egress queues to which the CICI Gateway subscribes.

- All communication must be performant and scalable across a variety of networks. This means that CIP must "stay home" or stay on premise where responses are timely and consistent.

- Any solution identified should avoid a specific implementation, but instead be described in general terms.

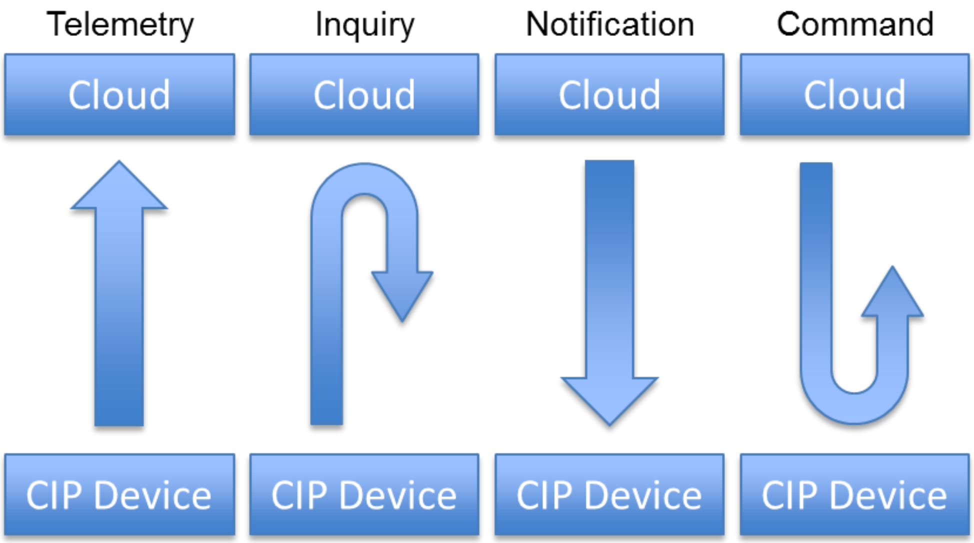
    - On one hand, this document avoids recommending implementation technologies. Instead, the focus is on D2C and C2D information exchange patterns (see below) common in the solution marketplace and relevant to the design of CICI and CICI-based solutions.

    - On the other hand, some technologies (AMQP, MQTT, JSON, etc.) are included in sample CICI architectures to illustrate how contemporary IIoT/Cloud integration are achieved and relate to information exchange patterns important to CICI.

Information Exchange Patterns

These Information Exchange Patterns generalize the logical patterns of communication that will occur between the cloud and gateways or devices. The figure below shows the four Information Exchange patterns and will be referenced in the use cases that are covered in the next section. For simplicity, the CiCi Gateway is not included in the diagram.

One way arrows imply no acknowledgement or response is expected. Return arrows imply expectation of a response.



Telemetry

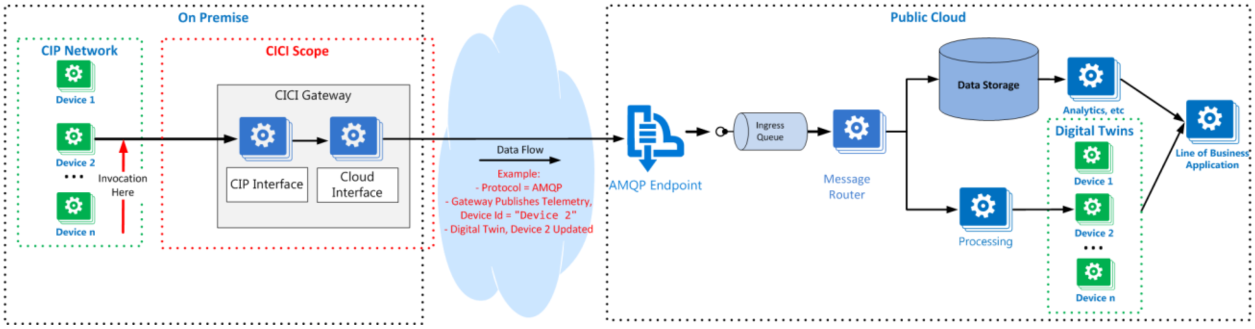
Telemetry is one-way, with information flowing that a device or gateway “volunteers” to a collecting service, either on a schedule or based on particular circumstances. This information represents the current or temporally aggregated state of the device or the state of its environment, like readings from sensors that are associated with it.

Telemetry is initiated by the CIP Devices based on a previous configuration or runtime subscription request. Data flows from the CIP Device to the CiCi Gateway and sent to the Cloud to be available for Cloud applications. Please note that security had been omitted for simplicity, but the connection to the cloud would have to set up before data would begin to flow following a Telemetry Information Exchange pattern.

The Telemetry exchange pattern is one way, so that the CIP Device does not expect a response from the Cloud application.

The following figure show the Telemetry Information Exchange pattern using the Reference Architecture. (Note: although there are many messaging technology choices, the following example reference architectures use AMQP as the message transport protocol and JSON as the message payload format.)

Reference Architecture: Telemetry



Inquiry

With Inquiries, the device or gateway solicits information about the state of the world beyond its own reach and based on its current needs; an inquiry is a singular request, but might also ask a service to supply ongoing updates about a particular information scope. A vehicle might supply a set of geo-coordinates for a route and ask for continuous traffic alert updates about particular route until it arrives at the destination.

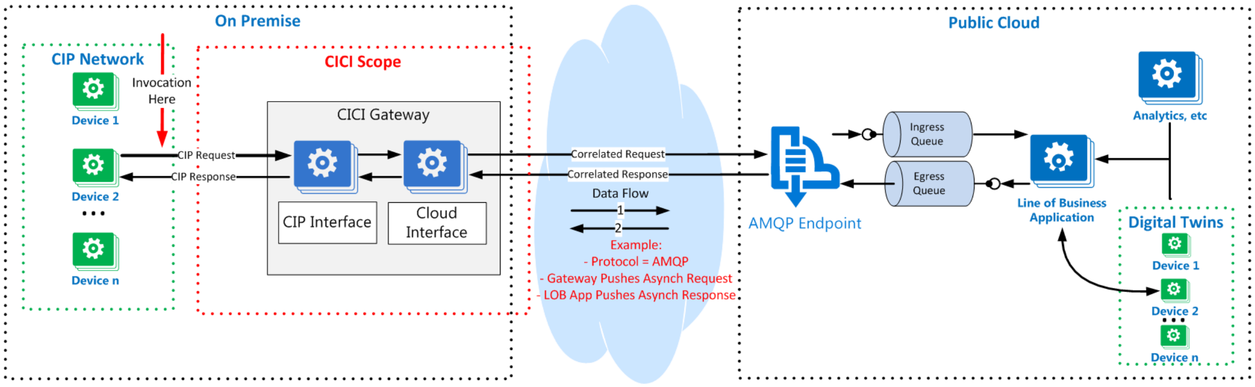
Inquiry is initiated by the CIP Device and flows through the CICI Gateway to the Cloud application via its messaging endpoint (AMQP in this example). The Cloud application receives the Inquiry message and creates a response. The Cloud app then sends the response message to the Egress queue to which the CICI Gateway is subscribed. The CICI Gateway routes the message to the correct Device.

The Inquiry exchange pattern differs from Telemetry in that a response or acknowledgement from the Cloud application is expected.

Of note is the use of Correlation IDs in the messaging between the CICI Gateway and the Cloud application. Correlation ID is an Enterprise Integration Design Pattern[7] used to relate a response to the correct request in asynchronous, non-guaranteed ordered communication scenarios.

The following figure show the Inquiry Information Exchange pattern using the Reference Architecture.

Reference Architecture: Inquiry



[7] Enterprise Integration Patterns, Hohpe, Woolf, Copyright 2004, Pearson Education, Inc, published by Addison-Wesley

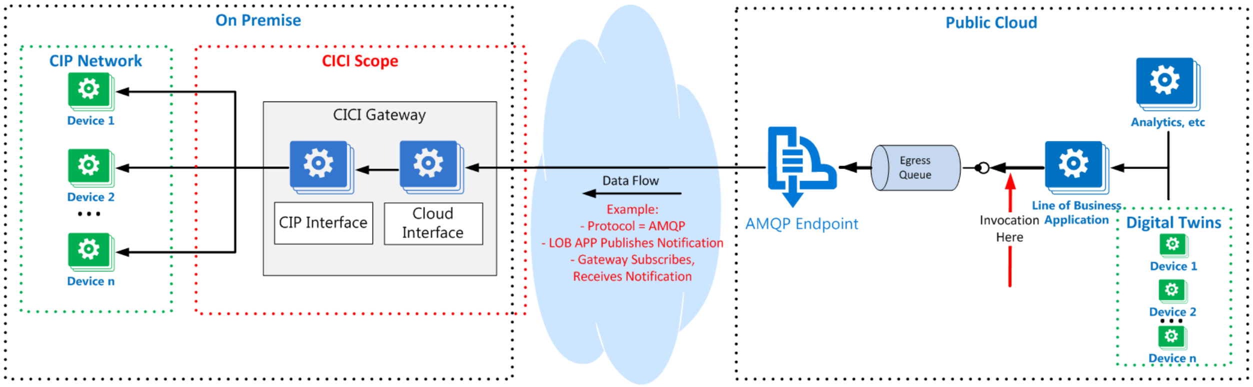
Notifications

Notifications are one-way, service-initiated messages that inform a device or a group of devices about some environmental state they'll otherwise not be aware of. Wind parks will be fed weather forecast information and cities may broadcast information about air pollution, suggesting fossil-fueled systems to throttle CO2 output or a vehicle may want to show weather or news alerts or text messages to the driver.

Notifications are initiated by the line-of-business Cloud applications. Cloud apps send Notification messages to the egress queue to which the CICI Gateway has subscribed. The CICI Gateway then routes the notification messages to the appropriate CIP Device or group of devices.

The Notification exchange pattern is the logical inverse of the Telemetry exchange pattern and, therefore, the Cloud application does not expect an acknowledgement or response from the device.

Reference Architecture: Notification



Commands

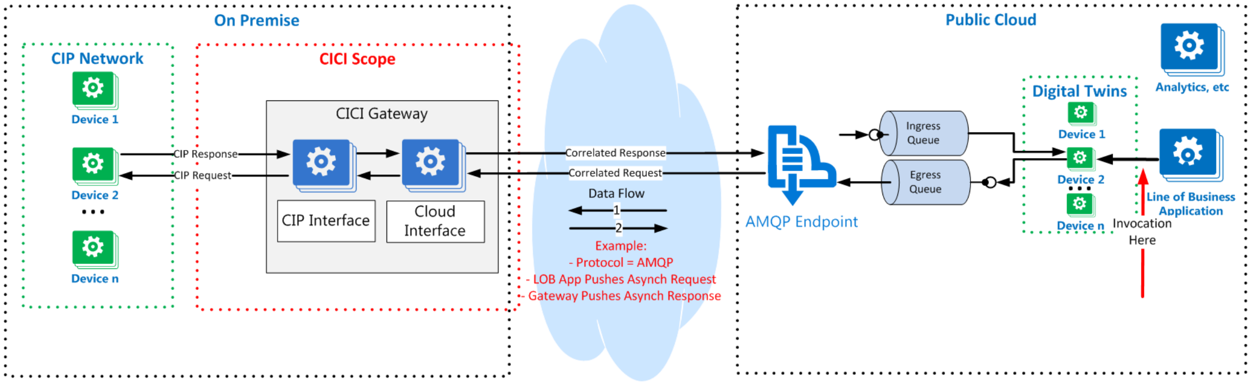
Commands are service-initiated instructions sent to the device. Commands can tell a device to provide information about its state, or to change the state of the device, including activities with effects on the physical world. That includes, for instance, sending a command from a smartphone app to unlock the doors of your vehicle, whereby the command first flows to an intermediating service and from there it's routed to the vehicle's onboard control system.

Command is initiated by the line-of-business Cloud application. Command messages are sent to the Digital Twin or directly to the egress command queue to which the CICI Gateway is subscribed. The CICI Gateway sends a synchronous request to the CIP Device and relays the response to the Cloud endpoint.

Command is the logical inverse of Inquiry. Therefore, the Cloud application expects a response from the Device.

Once again, the Correlation ID enterprise integration pattern can be leveraged for correct message logical routing and processing of responses.

Reference Architecture: Command



Use Cases

The SIG has decided to initially focus on the general area of Device Lifecycle Management. The following is a set of uses cases, grouped by different phases of a device's lifecycle. For each use case, a Cloud Information Exchange Pattern has been defined. The goal of the SIG is to develop or define a proposal for each of the Information Exchange Patterns, then validate their fitness for all identified use cases.

Commissioning

- Out-of-the-box definitions (Inquiry, Command)
- Cloud Registration / backend business setup (Inquiry, Command)
- On-boarding/Provisioning (Inquiry, Command)
- Context of device in application (Inquiry, Command)
- Control/Application loading (Inquiry, Notification, Command)

Operating

- Monitoring (Telemetry)
- Maintenance (Telemetry, Command)
- Calibration (Inquiry, Command)
- Diagnosis (Telemetry)
- Enable/Disable (Command)
- Optimization / Changing Parameters / Programs (Telemetry, Command)
- Software updates (Inquiry, Notification, Command)
- Device Replacement (Inquiry, Command)

Decommissioning

- Removing a device (Telemetry, Command)

Telemetry and CIP Devices

The communication pattern of Telemetry is the simplest pattern of the four patterns. It is also the most prevalent pattern in much of the “IIoT” conversations. These conversations all start with the premise that industrial devices generate useful data that needs to be sent to the cloud were it can be further analyzed. The following are a general set of steps that are necessary to enable telemetry. These steps are the basic steps and it is expected that additional details would be needed to form a complete solution.

1. Setup cloud for target device or gateway

   This step has many sub-steps. It includes setting up a unique identification of the device or gateway that is going to be sending data. It includes setting up what is going to happen to the data when it arrives on the cloud. It includes defining where the data will be stored and who might have access to the data.

2. Setup target device or gateway for cloud

   This step also has many sub-steps. It includes having a method of defining and securing the communication with the cloud. It may also need to have some additional information that is sent to help the cloud know how to process the data when it arrives on the cloud.

3. Select what and how often to send

   This step is a configuration or setup. It is necessary from either the cloud or on the device, to characterize what the data to be sent. The following are kinds of things that need to be selected or defined.

   a. Default values of how often to send the data
   b. System values of what data is to be sent for each type of device or gateway
   c. Device values specific data that is to be sent for a unique device

4. Connect (outbound messaging)

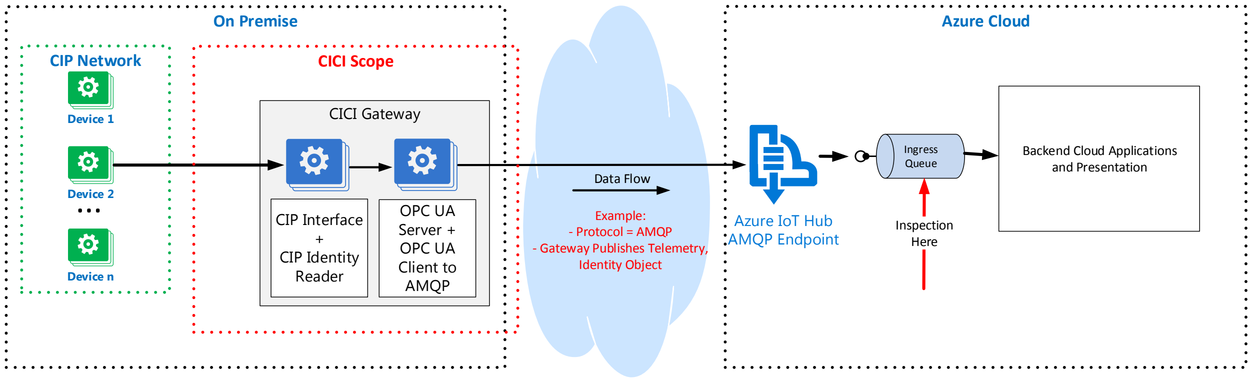
   This step is an action step that follows when the previous steps have been completed. The details about this assume a method to establish a secure connection that can route and process the data when it is communicated. This step might also define options for how a device or gateway responds when the connection is not available. The following are options that would be considered.

   a. Normal processing – fire and forget
   b. Abnormal condition handling – store and forward later

Proof-of-Concept

A simple Proof-of-Concept was defined that attempts to demonstrate a possible solution to the Telemetry pattern. The following diagram shows the components of the Proof-Of-Concept and explains the relationship to the steps of the Telemetry use case.

Reference Architecture: Proof-Of-Concept


On Premise
CIP Network
Device 1
Device 2
• • •
Device n
CICI Scope
CICI Gateway
CIP Interface + CIP Identity Reader
OPC UA Server + OPC UA Client to AMQP
Data Flow
Example:
- Protocol = AMQP
- Gateway Publishes Telemetry, Identity Object
Azure Cloud
Azure IoT Hub AMQP Endpoint
Ingress Queue
Inspection Here
Backend Cloud Applications and Presentation

CIP Network

- There are several CIP devices connected, each with a CIP Identity Object as required by ODVA standards.

CICI Gateway

- CIP Interface

  As a CIP device is discovered on the network, its CIP Identity Object contents are read and then inserted into an OPC UA Server.

- Cloud Interface
  - OPC UA Server

    An OPC UA Server represents the CIP Devices that have been discovered. Each CIP Device is represented as Node in the server’s namespace. All the attributes contained in each Identity Object are inserted under the CIP Device’s NodeID.

  - OPC UA Client to Azure IoT Hub (over AMPQ)

    An OPC UA Client reads the server’s namespace, starting with the root node, selecting the CIP Identity data and inserting into an AMQP message. Those AMQP messages are sent to an Azure IoT Hub endpoint.

Data Flow

- AMQP Messaging

Public Cloud

- Microsoft’s Azure IoT Hub (receiving AMQP)
  - A tool is used to inspect the AMQP messages as they are received on the AMQP endpoint for further processing.

Conclusion - Next steps

This whitepaper covered some new technologies that are being introduced into the industrial automation marketplace. It introduced a communication framework that the Common Industrial Cloud Interface SIG intends to use to guide its efforts. It provided a “first pass” definition of one of the communication patterns and an example of how that communication pattern could be realized. It can be easily observed that the Common Industrial Cloud Interface SIG is just getting started. There is a lot of ground to be covered to completely define the remaining three communication patterns at a high level. The next step will be to fill in more details, referencing all the technologies and standards available and keeping in mind the guiding principles enumerated.

The goal of the Common Industrial Cloud Interface SIG is to leverage cloud technologies to provide consumers and producers the most value throughout the entire lifecycle of devices that use ODVA licensed technologies.

If you would like more information or want to contribute, please consider joining the SIG.

****

The ideas, opinions, and recommendations expressed herein are intended to describe concepts of the author(s) for the possible use of ODVA technologies and do not reflect the ideas, opinions, and recommendation of ODVA per se. Because ODVA technologies may be applied in many diverse situations and in conjunction with products and systems from multiple vendors, the reader and those responsible for specifying ODVA networks must determine for themselves the suitability and the suitability of ideas, opinions, and recommendations expressed herein for intended use.