Gartner Research

Reference Architecture for Integrating OT and Modern IT

Paul DeBeasi

31 July 2023

Gartner®

Gartner®

Reference Architecture for Integrating OT and Modern IT

Published 31 July 2023 - ID G00792965 - 36 min read

By Analyst(s): Paul DeBeasi

Initiatives: Infrastructure for Technical Professionals

Organizations are striving to modernize their industrial operations, but are hampered by aging architectures. This document helps I&O technical professionals overcome this challenge by describing an IT-OT reference architecture, explaining its components, and analyzing its strengths and weaknesses.

Overview

Key Findings

- The hierarchical levels and point-to-point connections of the Purdue Model require developers to integrate a disjointed set of incompatible, vendor-defined APIs. The result is a proliferation of incompatible interfaces that make it difficult for developers to discover, access and analyze business data trapped in disparate operational technology (OT) systems.

- The Purdue Model assumes that architects isolate OT systems from IT systems. The IT-OT isolation creates zones of trust that protect OT assets. However, the Internet of Things has blurred IT-OT boundaries, broken down zones of trust and exposed attack surfaces.

- A new industrial IT-OT architecture is emerging to improve data accessibility and reduce asset vulnerability. It will be built upon an event-centric integration pattern that will use MQTT brokers, the Sparkplug B standard and a Unified Namespace (UNS) design strategy. It will also use OT asset discovery and protection platforms.

- The Sparkplug B standard provides important extensions to MQTT (e.g., report by exception [RBE]) but has technical limitations (e.g., inability to handle command/response use cases). The UNS design strategy improves access to OT data, but requires collaboration across many different roles and teams.

Recommendations

- Modernize industrial edge architecture by transitioning from a Purdue Model toward an event-centric integration pattern. Deploy event brokers that support MQTT 5.0 as the backbone of your design. Ensure that the brokers provide Sparkplug B Aware compliance.
- Use a cyber-physical security protection platform (CPS-PP) to discover your OT assets before defining your namespace data hierarchy. Use the same CPS-PP to protect your OT assets and to integrate with IT security tools, such as security information and event management (SIEM).
- Define a MQTT/Sparkplug namespace data hierarchy for your OT assets by using the ISA-95 Part 2 equipment model. Pilot your MQTT/Sparkplug namespace in one facility before deploying a multifacility solution. Test capabilities such as data auto discovery, RBE, reliability and scalability.

Strategic Planning Assumptions

By 2025, more than 50% of enterprise-managed data will be created and processed outside the data center or cloud.

Through 2025, 70% of companies will deploy cyber-physical systems protection platforms as the first step in their asset-centric security journey.

Analysis

Data pipelines in industries such as manufacturing, energy and utilities have become critical to supporting diverse, complex and mission-critical business processes. But many organizations struggle to find, distribute and analyze their OT data. Even when OT systems do share data, the data is often unstructured, fragmented and real-time. Pressure to deliver faster results, higher quality and greater resiliency is driving organizations to consider how they can integrate OT with IT.

Threats to OT systems are on the rise. The Purdue Model used to be the gold standard for industrial security. The model created trust zones by segmenting industrial systems into functional layers. However, as OT systems became more connected, they created holes in the trust zones and exposed attack surfaces. This made them attractive targets for ransomware and malware. These attacks can lead to asset damage, system downtime and production losses. Even worse, they may cause harm to operations staff (e.g., by overriding safety controls in a chemical plant). Many organizations do not have an accurate inventory of their deployed OT assets. As a result, they are unable to secure assets they don't know they possess.

Some organizations are attempting to address these issues. They are deploying modern IT technologies, such as MQTT event brokers and security solutions that automate OT asset discovery and protection. Some are embracing the Sparkplug B standard and a hierarchical, ISA-95-inspired data model called the Unified Namespace. They are also deploying platforms such as data hubs and Industrial Internet of Things (IIoT) platforms. Most organizations are at the early stages of this industrial transformation, and thus have limited experience making these changes.

How should technical professionals architect a solution that integrates OT with IT?

This document presents an emerging industrial IT-OT reference architecture. We expect this architecture will evolve over time as organizations gain experience and make necessary refinements.

The document is structured as follows:

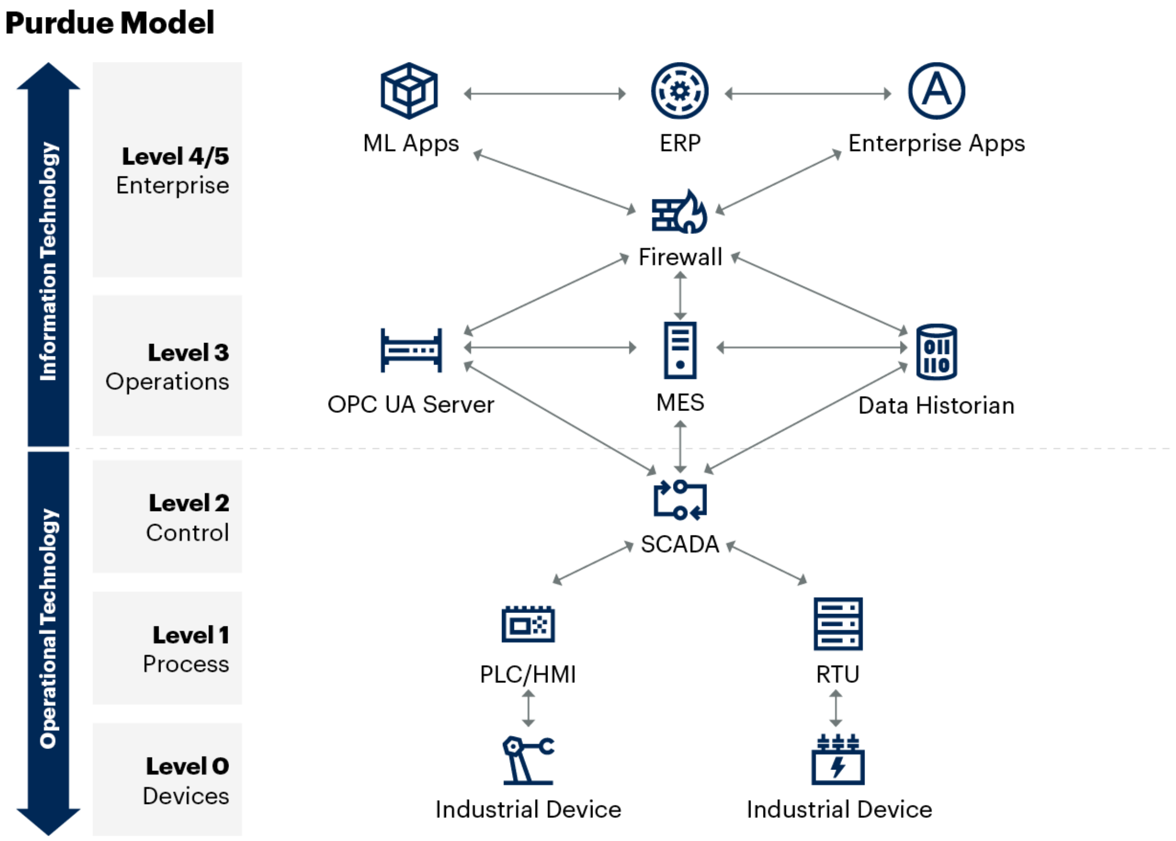
- Purdue Model — An Outmoded Architecture
- IT-OT Reference Architecture — An Event-Centric Model
- Reference Architecture Components
- Unified Namespace
- IT-OT Design Examples
- Strengths and Weaknesses

- Guidance

Purdue Model — An Outmoded Architecture

Many OT architectures were designed using the 1990s-era Purdue Enterprise Reference Architecture (also known as the Purdue Model or ISA-95). This model segments OT networks into functional levels and uses point-to-point connections within, and between, each level to share data across the OT domain (see Figure 1). As a result, developers must integrate a disjointed set of incompatible, vendor-defined APIs to create these connections and access business data trapped in disparate OT systems. Operators must manually define new data sources (e.g., sensor tags) and manually integrate that data into application software (e.g., supervisory control and data acquisition [SCADA] systems). The result is a proliferation of incompatible interfaces that make it challenging for developers to support the sharing and analysis of data using modern IT software, such as stream processing and machine learning.

Figure 1: Purdue Model



ERP = Enterprise Resource Planning; HMI = Human-Machine Interface; MES = Manufacturing Execution System; ML = Machine Learning; OPC UA = Open Platform Communication Unified Architecture; PLC = Programmable Logic Controller; RTU = Remote Terminal Unit; SCADA = Supervisory Control and Data Acquisition

Source: Gartner

792965_C

The Purdue model originated as a generalized distributed control architecture. The model designers never intended to create an industrial security architecture. They assumed that OT systems would remain isolated from IT systems and thus remain secure. Designers used firewalls to form trust zones between model layers. This approach became the gold standard for industrial security.

However, the Internet of Things (IoT) changed all that. For example, modern sensors and devices can generate data streams and send those streams directly to the cloud, bypassing trust zones. As a result, the IT-OT boundaries have blurred. Industrial architects are reluctantly coming to the realization that they must apply zero-trust security principles to their OT assets.

IT-OT Reference Architecture — An Event-Centric Model

A new industrial IT-OT architecture is emerging. It represents the next step in the evolution from the hierarchical Purdue Model toward a distributed, interconnected model. Various organizations have proposed new industrial models, such as the Industrial internet Reference Architecture (IIRA) [1] and Reference Architectural Model Industrie 4.0 (RAMI 4.0). [2] These models are broad in scope. They describe architectural concepts, viewpoints, stakeholders and relationships. But they do not describe a specific solution architecture.

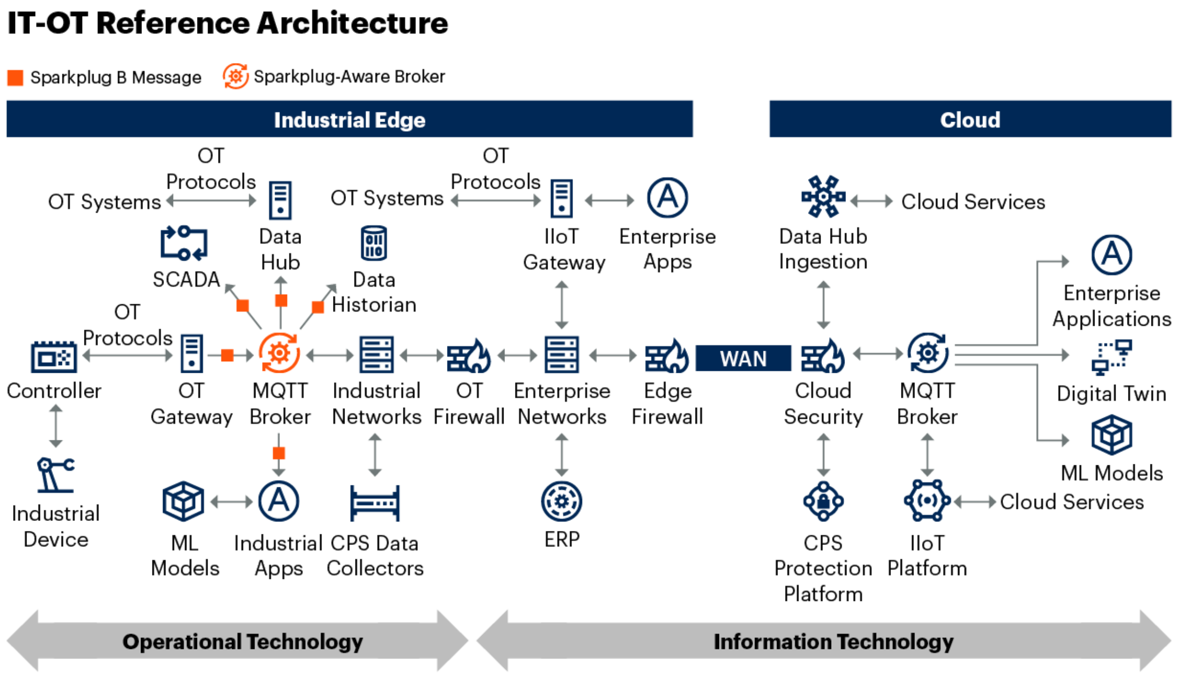
The reference architecture described herein proposes a specific solution architecture. The primary value of this architecture is that it improves the ability of industrial organizations to discover, secure, share and analyze their OT data. The architecture integrates the industrial edge (aka "the core") with cloud computing (see Figure 2).

The industrial edge represents the physical plant. It contains industrial devices (e.g., robotic arms), OT systems (e.g., SCADA systems), and industrial networks (e.g., Modbus). The industrial edge also includes IT systems (e.g., machine learning inference engines), enterprise applications (e.g., ERP), and enterprise networks (e.g., Wi-Fi).

The cloud represents the applications running in the cloud. Some of these applications (e.g., the IIoT platform and CPS-PP) interact with the industrial edge systems (e.g., the IIoT gateway and CPS data collectors). Other applications analyze edge data that has been transferred to the cloud (e.g., ML models and digital twins).

The industrial edge connects to the cloud over a secure connection using a variety of technologies (IPsec, VPN, SD-WAN, etc.). The edge-to-cloud transport protocols also vary considerably. They are dependent upon which edge and cloud applications are communicating with each other and which protocols they use (HTTPS, MQTT, FTP, etc.).

Figure 2: IT-OT Reference Architecture



Source: Gartner

CPS = Cyber-Physical System; ERP = Enterprise Resource Planning; HMI = Human-Machine Interface; IIoT = Industrial Internet Of Things; MQTT = MQ Telemetry Transport; MES = Manufacturing Execution System; ML = Machine Learning; OPC UA = Open Platform Communication Unified Architecture; OT = Operational Technology; SCADA = Supervisory Control and Data Acquisition; WAN = Wide-Area Network

792965_C

Reference Architecture Components

The industrial reference architecture contains the following key components:

- **MQTT Broker:** An MQTT broker forms the basis of an event-centric integration pattern. MQTT is a lightweight, standards-based, publish-subscribe messaging protocol. The MQTT broker moves messages from event producers (e.g., industrial devices) to event consumers (e.g., SCADA systems). Events are unidirectional, traveling from publisher to subscriber.

- **Data Hub:** The data hub accesses OT data at the edge and shares it with applications in the cloud. The hub normalizes data formats and maintains data flows between applications. Data hubs use a variety of methods to transfer data from the edge to the cloud (Apache Kafka, MQTT messages, file transfer, etc.)

- **CPS Protection Platforms and Data Collectors:** CPS-PPs discover and protect cyber-physical assets. CPS data collectors send asset data to the CPS protection platform, located in the cloud or at the edge.

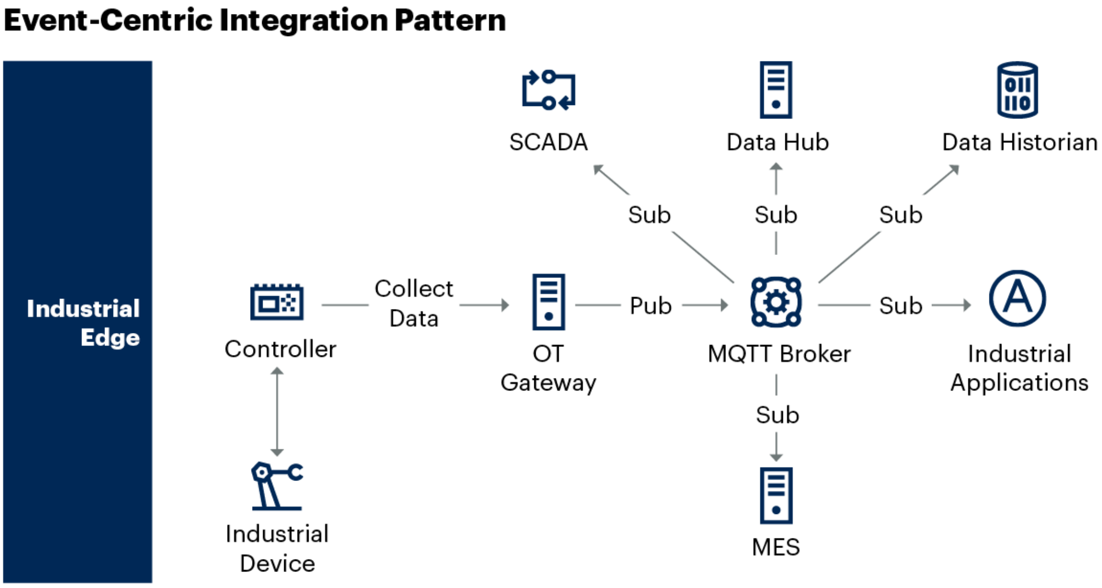
- **IIoT Platform and Gateway:** IIoT platforms support IT-OT integration at scale. IIoT platforms orchestrate siloed data sources to improve insights and actions across a heterogeneous asset. An IIoT gateway communicates with an IIoT platform using event protocols (e.g., MQTT events), management protocols (e.g., digital twin state updates) and control protocols (e.g., deploy, install, start a containerized application from the cloud to the edge).

- **Sparkplug B:** Sparkplug B is an open-source specification that defines an MQTT topic namespace, state management, and the payload encoding scheme.

MQTT Broker

The proliferation of event-centric architectures and stream processing technologies have enabled new integration patterns that complement many of the familiar data integration methods (e.g., extract transform and load). One of the more common patterns is the event-centric integration pattern. This pattern forms the foundation of the industrial reference architecture.

Figure 3: Event-Centric Integration Pattern



Source: Gartner

SCADA = Supervisory Control and Data Acquisition; MQTT = MQ Telemetry Transport; MES = Manufacturing Execution System; OT = Operational Technology

792965_C

An event-centric integration pattern delivers events, or streams of events, to the endpoints that consume them. Events travel from source endpoint (aka publisher) to sink endpoint (aka subscriber). Different sources can publish to the same event type or topic, and many sinks can subscribe to those events without explicitly knowing who the publishers or the subscribers are. Event-centric integration payloads may involve either individual events or an unbounded stream of events.

For example, in Figure 3, an edge node collects industrial data from a robotic arm controller. The edge node creates an MQTT event message, copies the industrial data into the MQTT payload and transmits the MQTT message to the MQTT broker using a MQTT topic. The MQTT broker forwards the message to the five applications that subscribed to that MQTT topic (refer to the MQTT Broker section for a description of topics). In this figure, the edge node is the source/publisher, and the five applications are the sinks/subscribers.

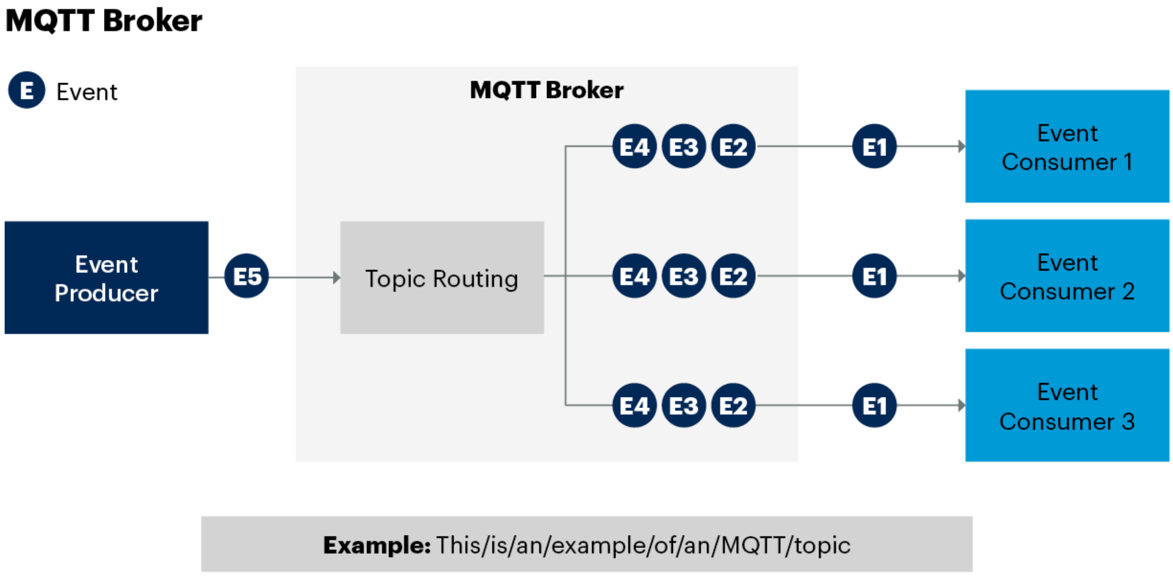
Compare Figures 1 and 3. In Figure 1, the SCADA system communicates with the data historian over a point-to-point connection using a vendor-defined API. Every new SCADA-to-device integration requires a new point-to-point connection. In Figure 3, the SCADA system communicates with the data historian using a publish-subscribe protocol via the MQTT broker. The SCADA system needs only one connection to the MQTT broker, regardless of the number of interconnected devices. This approach is easier to scale and maintain than the Purdue Model approach. For more information on event-centric integration patterns, refer to:

- Streamline Integration by Choosing Between Data-, Event- or Application-Centric Styles
- Essential Patterns for Data-, Event- and Application-Centric Integration and Composition
- Essential Patterns for Event-Driven and Streaming Architectures

A key component of the event-centric integration pattern is the event broker. The broker moves messages from event producers to event consumers. The event broker decouples producers and consumers in time, space and synchronization. The decoupling enables technical professionals to add producers and consumers without impacting the rest of the system, thus improving design flexibility and system scalability.

Brokers use topic routing logic to route events from producer to consumer (see Figure 4). The topic format is dependent on the protocol supported by the event broker. For instance, the MQTT protocol uses a forward-slash delimited string as the topic format (“This/is/an/example/of/an/MQTT/topic”). A broker uses a topic header to route events to a subscriber, similar to how an IP router uses an IP multicast address to route packets. The IP router uses a “subscription” to the multicast group to determine which interfaces it must route the packet to. The MQTT broker routing logic is basic pattern matching. An MQTT subscriber provides a string that can be a specific topic, or use wildcards to subscribe to matching messages.

Figure 4: MQTT Broker



Source: Gartner
MQTT = MQ Telemetry Transport
792965_C

MQTT is a lightweight, standards-based, publish-subscribe messaging protocol. It was originally developed by IBM for real-time SCADA systems communication over very small aperture terminal satellite networks. It is now an OASIS standard. The MQTT standard uses the term “MQTT server,” but we use the term “MQTT broker” for document clarity and consistency. MQTT brokers are widely available as managed cloud services, open-source projects and commercial applications. Vendors have begun to integrate MQTT clients into their industrial products. MQTT runs over TCP/IP and WebSockets. Thus, the MQTT header and payload can be encrypted using transport layer security (TLS).

Most MQTT brokers support versions 3.1.1 and 5.0. Gartner recommends using brokers that support 5.0 onward, because it has many enhancements that improve MQTT operations and security, including:

- **User Properties:** Improves extensibility of MQTT by allowing developers to append UTF-8 string key-value pairs to almost any MQTT packet.
- **Reason Codes:** Many packets now contain error reason codes.
- **Enhanced Authentication:** Enables challenge/response authentication, including mutual authentication.

> MQTT brokers are a key component in a modern IT-OT architecture.

Note that 5.0 client-to-client features (e.g., user properties) require a 5.0 client for both publisher and subscriber, whereas 5.0 client-to-broker features (e.g., enhanced authentication) do not. MQTT brokers must support a mix of MQTT 3 and MQTT 5.0 clients to enable a graceful migration to MQTT 5.0. Check with your MQTT supplier to ensure this is the case.

Gartner recommends placing the MQTT broker behind the OT firewall, because doing so requires only one port to be open in the OT firewall, regardless of the number of publishers and subscribers (see Figure 2). The broker enables communication between OT systems and enterprise applications.

The broker uses access control technology to explicitly control which clients can communicate with the broker, as part of a move to zero trust. MQTT 3.0 uses the simple exchange of username/password for client authentication. MQTT 5.0 uses the new AUTH packet to authenticate a client by presenting a challenge that the client must respond to. This enables organizations to use security protocols such as Kerberos or salted challenge response authentication mechanism (SCRAM).

There are many stand-alone MQTT brokers to choose from that support on-premises deployment, including Eclipse Mosquitto, EQMX, HiveMQ and RabbitMQ. Some of these vendors provide support for functionality, such as high-availability configurations and broker-to-broker bridging. Some brokers also support multiple protocols (e.g., MQTT and AMQP).

Some hyperscaler IoT platform vendors have integrated event brokers into their IoT gateway software (e.g., Amazon Web Services [AWS] IoT Greengrass, Microsoft Azure IoT Edge). Data hub vendors have also embedded event brokers into their hubs (e.g., HighByte, Cogent). If you choose to deploy multiple event brokers from different vendors, then you must ensure the brokers support the same messaging protocol, topic structure (see Unified Namespace) and message encoding scheme (see Sparkplug B). For additional event broker information, refer to Choosing Event Brokers: The Foundation of Your Event-Driven Architecture.

For additional MQTT information, refer to the following:

- MQTT Essentials: The Ultimate Guide to MQTT for Beginner and Experts
- MQTT 5 Essentials: A Technical Deep Dive Into New MQTT 5 Features
- Differences Between 3.1.1 and 5.0
- Introduction to MQTT Security Mechanisms
- Understanding the MQTT Protocol Packet Structure

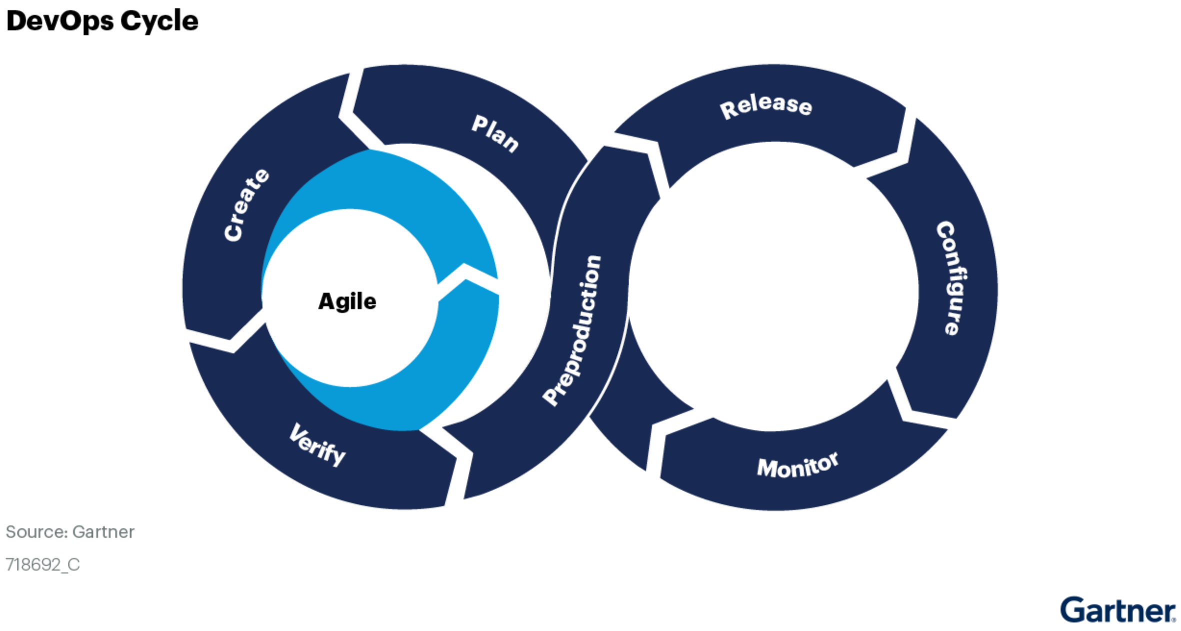
Data Hub

Data and analytics teams cannot achieve the project deployment speed they desire because too many roles, too much complexity and constantly shifting requirements make it difficult. In most organizations, this complexity is exacerbated by limited or inconsistent coordination across the roles involved in building, deploying and maintaining industrial data pipelines. DataOps techniques can address these challenges through a more agile, collaborative and change-friendly approach to building and managing data pipelines.

DataOps is a collaborative data management practice focused on improving the communication, integration and automation of data flows between data managers and data consumers across an organization.

DataOps is about a more efficient way of working when delivering data and analytics solutions. Applying techniques adapted from the DevOps concepts (see Figure 5), which many organizations have leveraged in implementing applications, better communication and tighter collaboration results in faster deployments and greater effectiveness in reacting to change postdeployment. With the increasing awareness of DataOps concepts and terminology, organizations are looking for the best ways to introduce these ideas into their operations teams.

Figure 5: DevOps Cycle



Source: Gartner
718692_C

For many, DataOps represents a massive shift in approach, raising substantial change management concerns. Leaders who seek to increase their teams' effectiveness must identify effective ways to gradually introduce DataOps techniques into data and analytics solution delivery methodologies. For more information on DataOps, refer to

- How to Operationalize Data Workloads

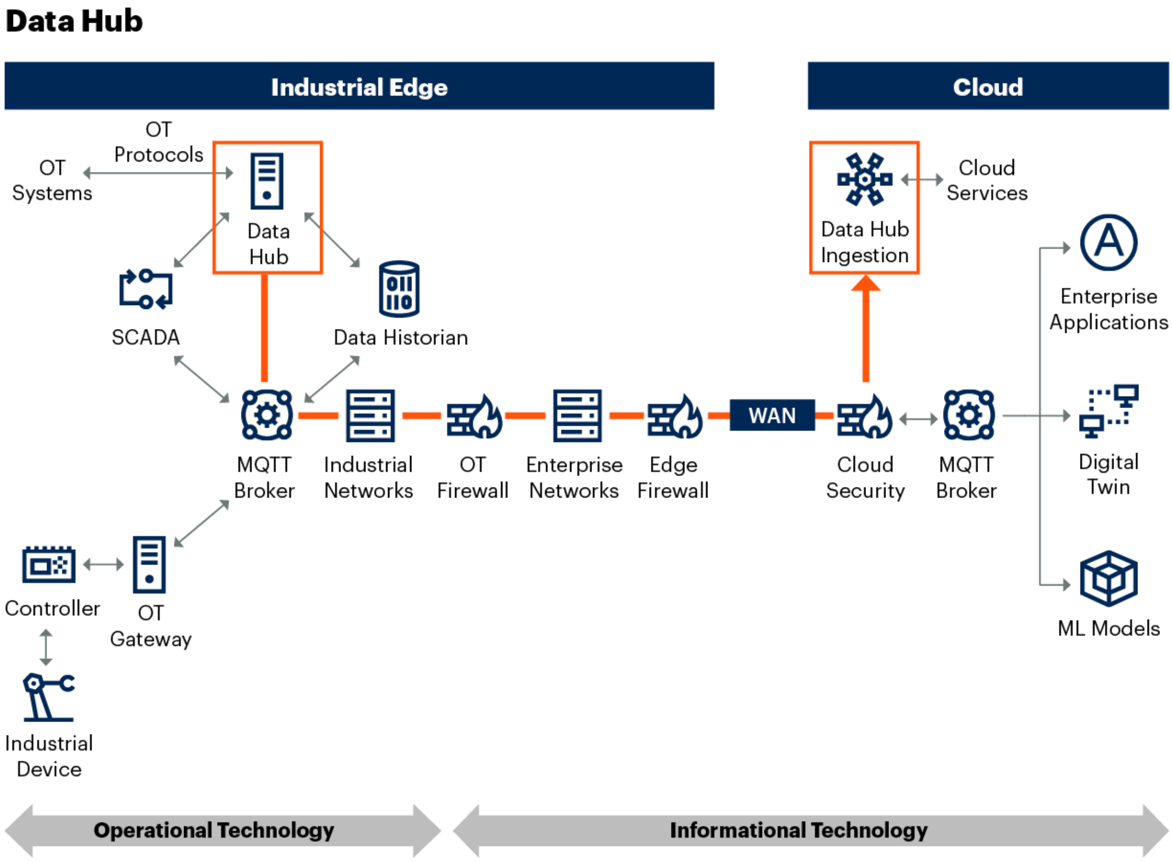
- Data and Analytics Essentials: DataOps
- Market Guide for DataOps Tools

DataOps practices rely upon having connections among systems, people and things as a means of sharing data for insights, innovation and competitive advantage. Enterprises often meet this need by two approaches: connecting applications and data sources via point-to-point interfaces (e.g., the Purdue Model) and centralizing as much data as possible in a single system (e.g., a cloud database). Both approaches eventually become costly, inflexible and run into scalability issues as data volume grows.

Integration of a data hub (see Figure 6) enables improved data sharing and integration via more consistent, scalable and well-governed data flow. A data hub can deliver benefits, including:

- Increased operational effectiveness through consistent governance of data and analytics across sets of diverse endpoints
- Reductions in cost and complexity compared with point-to-point integration
- Improvements in understanding and trust of critical data across process and organization boundaries
- Alignment of data and analytics initiatives focused on governance and sharing of critical data
- Filtering and sharing of industrial edge data with cloud services

Figure 6: Data Hub



Source: Gartner

ML = Machine Learning; MQTT = MQ Telemetry Transport; OT = Operational Technology; SCADA = Supervisory Control and Data Acquisition; WAN = Wide-Area Network

792965_C

Data hubs from vendors such as HighByte, Cogent and Cybus provide connections to a variety of OT systems and protocols (e.g., OPC/UA, Modbus), cloud services (e.g., Amazon Kinesis, Azure Event Hubs, Google Cloud Pub/Sub), pub/sub protocols (e.g., MQTT, Sparkplug B) and a growling list of connectors. The data hub connector is the method by which organizations provide integration with their existing systems. For additional information on data hubs, refer to Data and Analytics Essentials: Data Hubs and Implementing the Technical Architecture for Master Data Management.

CPS Protection Platform and Data Collectors

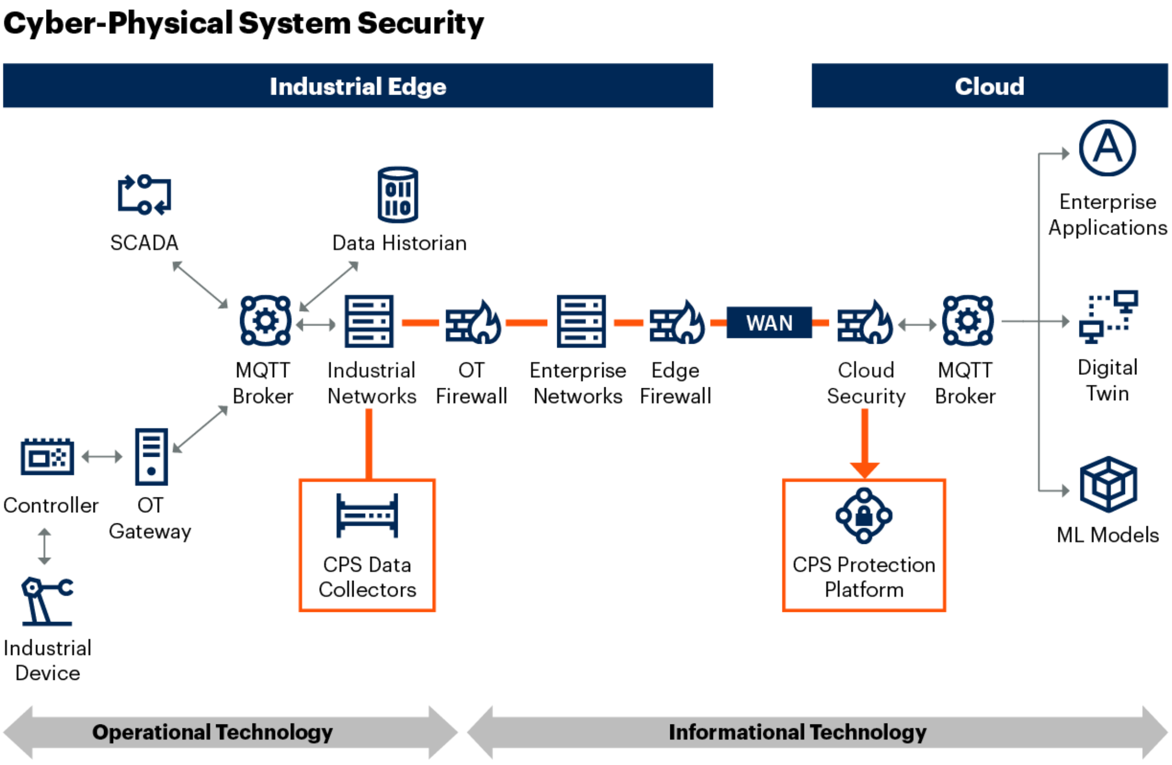
Gartner defines cyber-physical systems (CPS) as “engineered systems that orchestrate sensing, computation, control, networking and analytics to interact with the physical world (including humans). They enable safe, real-time, secure, reliable, resilient and adaptable performance.” As more organizations connect CPS’s to each other, to the enterprise, to workers and to OEMs, they expand the attack surface, and increase their security risk. In 2023, the European Union issued the Network and Information Security 2 (NIS2) Directive to respond to the growing threats posed by digitalization and the surge in cyberattacks. [4] A recent Gartner survey found that “internal and insurance audits also usually find large cybersecurity gaps, and [security and risk management] leaders increasingly face challenges to update their governance efforts to deal with the situation.” (For more information, see CPS Security Governance — Best Practices From the Front Lines.)

Over the past few years, CPS-PPs have emerged to help security leaders inventory their OT assets. The reason CPS-PPs have risen to such prominence in the OT world is because they lead with asset discovery. CPS-PPs may not find all OT assets, due to air-gapped equipment and firewalls. However, it is not unusual for organizations to end up with an inventory that shows they have up to 75% more assets than they knew about. Not only do CPS-PPs provide a front end to the security process, they are also becoming a back end by capturing telemetry, utilization and operational data. This information has value to the engineering team.

CPS platform features initially focused on asset discovery, visibility and network topology. However, as the products evolved, they became CPS protection platforms. The platforms increasingly interoperate with other security tools, such as SIEM, or security orchestration, automation and response (SOAR) solutions.

Organizations deploy the CPS platform in the cloud (as in Figure 7) or as an on-premises appliance. The CPS data collectors reside at the industrial edge, often behind the OT firewall. The collectors perform passive and active monitoring of OT assets. The collectors forward asset information to the CPS-PP, as illustrated by the orange arrow in Figure 7. There are many different types of data collectors. They include passive discovery methods (e.g., networking monitoring using Switched Port Analyzer ports), operating system clients (e.g., Microsoft Defender for Endpoint), network infrastructure polling (e.g., DHCP, DNS) and OT protocol scanning (e.g., Modbus).

Figure 7. Cyber-Physical System Security



Source: Gartner

ML = Machine Learning; MQTT = MQ Telemetry Transport; OT = Operational Technology; SCADA = Supervisory Control and Data Acquisition; WAN = Wide-Area Network

792965_C

Sample vendors include Armis, Claroty, Dragos, Forescout, Fortinet, Microsoft, Nozomi Networks, Ordr, SCADAfence and Xage Security. For more information on CPS-PPs, see:

- Innovation Insight for Cyber-Physical Systems Protection Platforms
- Predicts 2023: Cyber-Physical Systems Security — Beyond Asset Discovery
- Market Guide for Operational Technology Security

The IT-OT reference architecture incorporates security capabilities in addition to those provided by the CPS-PP. These include:

- Isolation of the OT network from the IT network via an OT firewall. By default, port 8883 is open for event messages (e.g., MQTT).

- Isolation of the IT network from the WAN via an edge firewall. By default, port 8883 is open for event messages.
- Protection of edge-to-cloud communication via VPN technology.
- Assurance that edge appliances (e.g., edge firewall, IIoT gateway, CPS data collectors) are trusted devices via an X.509 certificate.
- Control of client-to-broker communication via access control technology. Refer to the MQTT Broker section for details.
- Encryption of MQTT messages via use of TLS. By default, port 8883 is open for event messages. Refer to Section 5 of the MQTT specification for additional information.

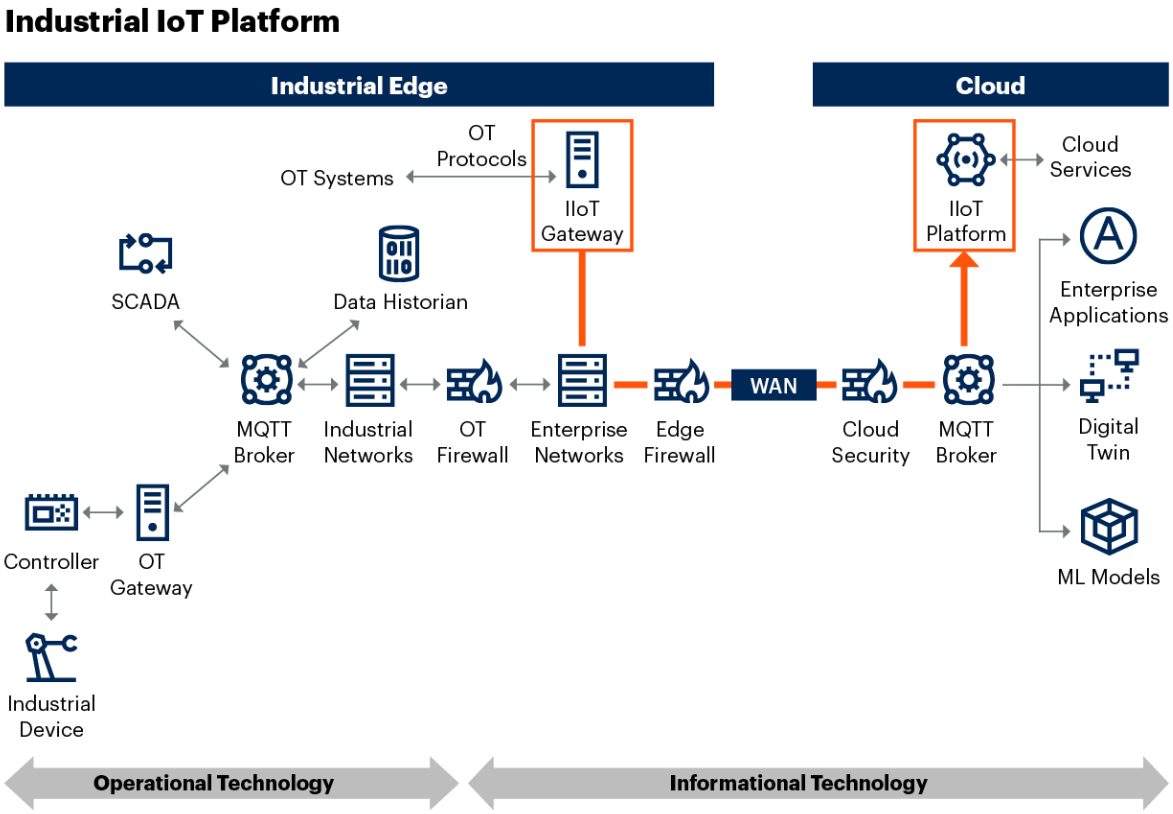
Industrial IoT Platforms

IIoT platforms support IT-OT integration at scale. The IIoT platform is differentiated from older-generation OT equipment by its ability to collect higher volumes of time series and/or low-latency complex machine data from networked IoT endpoints in a cost-effective manner. It is engineered to provide security, safety, sustainability, automation, device management, and other mission-critical objectives associated with industrial assets.

The IIoT platform may be consumed as a technology suite, a general-purpose application platform or both in combination. It can be deployed in the cloud (as in Figure 8), at the edge or in data centers. In December 2022, Gartner's Magic Quadrant for Global Industrial IoT Platforms analyzed approximately 1,900 industrial IoT deployments. The analysis found that 40% of enterprises preferred a hybrid (cloud and edge) IIoT platform deployment model, closely followed by cloud only at 36%, and the remainder using edge only at 24%.

The IIoT platform communicates with the IIoT gateway to access historically siloed data sources and improve insights across heterogeneous asset groups. The IIoT gateway provides event processing, workload orchestration and OT system integration at the edge. The IIoT platform deploys workloads (e.g., ML inference, stream analytics) onto the IIoT gateway. The gateway runs these workloads and forwards data to the IIoT platform. The orange arrow in Figure 8 illustrates data being forwarded from the gateway to the IIoT platform. IIoT vendors usually provide the gateway software as a GitHub open-source project (e.g., Azure IoT Edge, AWS IoT Greengrass, thin-edge.io).

Figure 8: Industrial IoT Platform



Source: Gartner

ML = Machine Learning; MQTT = MQ Telemetry Transport; OT = Operational Technology; SCADA = Supervisory Control and Data Acquisition; WAN = Wide-Area Network

792965_C

Sample vendors include ABB, AWS, Envision Digital, Hitachi, Microsoft, Siemens and Software AG. For additional information, refer to Architect IoT Using the Gartner Reference Model, and Emerging Technologies: AI-Enabled IoT.

Sparkplug B

The MQTT standard does not define a topic structure, nor does it define a payload encoding scheme. Sparkplug B is an open-source specification, managed by the Eclipse Foundation, that addresses both needs to improve interoperability. In addition, the specification makes use of MQTT's native continuous sessions awareness capability. The Eclipse Foundation released Sparkplug B version 3 in November 2022.

Sparkplug B has several key design principles:

- **Pub/Sub Edge Node:** The Sparkplug B Edge node is an MQTT pub/sub client. The Sparkplug B edge node acts as a gateway to older generation devices that do not support MQTT or Sparkplug B (see Figure 9).

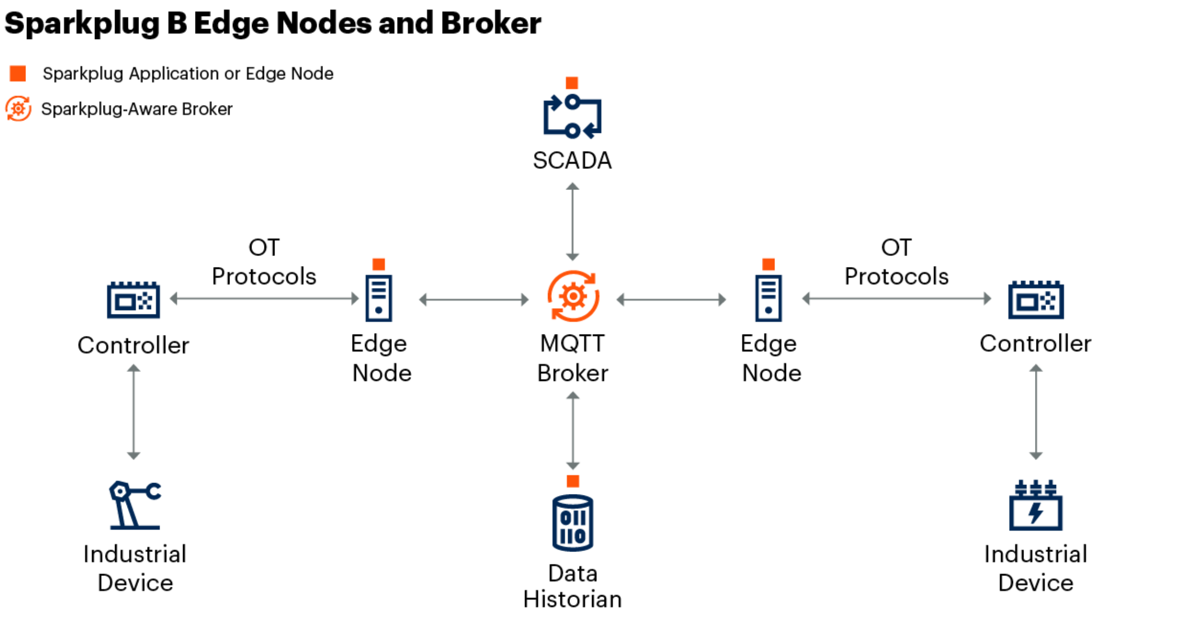
- **Continuous-State Awareness:** Sparkplug B defines BIRTH and DEATH messages and makes use of the MQTT "Will Message" to maintain session awareness. This is important because it eliminates the need for applications to poll an edge node to determine session state.

- **Report by Exception:** Edge nodes only publish data when values change in the devices managed by the edge node. This is much more efficient than having an application poll the edge node to discover data value changes.

- **Birth and Death Messages:** The NBIRTH message is the first message an edge node publishes. It includes the definition and current value for every metric managed by the edge node. This message enables applications to dynamically and efficiently discover data. This is powerful because it eliminates the need for static device and tag configuration. This is analogous to how a router dynamically discovers networks, instead of requiring static network configuration. The NDEATH message is the final message an edge node will publish prior to intentionally disconnecting from the broker. The NDEATH message notifies applications that the edge node is offline, and all metrics managed by the edge node are stale.

- **Google Protocol Buffer Encoding:** Sparkplug B uses Google protocol buffers for data encoding. Tests have shown that Google protocol buffer encoding can reduce network bandwidth consumption by more than 75%, compared to Modbus. [5]

Figure 9: Sparkplug B Edge Nodes and Broker



Source: Gartner

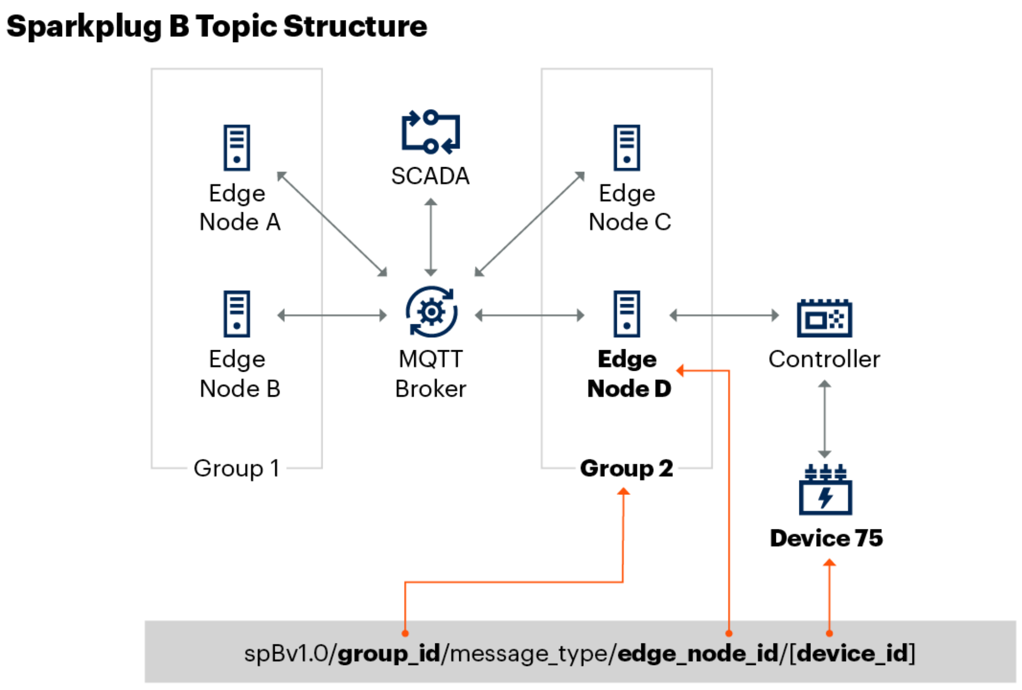
MQTT = MQ Telemetry Transport; OT = Operational Technology; SCADA = Supervisory Control and Data Acquisition

792965_C

Sparkplug B defines a topic structure for MQTT messages (see Figure 10) that includes the following components:

- **spBv1.0:** Every Sparkplug B topic must begin with these characters.
- **Group_id:** Provides a logical grouping of Sparkplug Edge Nodes
- **Message_type:** Indicates how to process the MQTT payload (e.g., NBIRTH, NDEATH, DDATA, DCMD).
- **Edge_node_id:** Identifies the specific edge node.
- **Device_id:** Identifies the device connected (logically or physically) to the edge node. This is an optional element.

Figure 10: Sparkplug B Topic Structure



Source: Gartner

MQTT = MQ Telemetry Transport; SCADA = Supervisory Control and Data Acquisition

792965_C

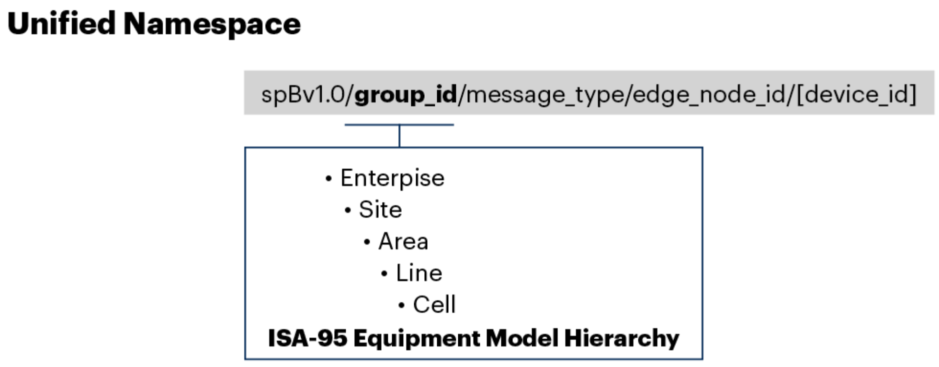
Sparkplug B defines “Sparkplug-compliant” and “Sparkplug-aware” MQTT servers. Generally, a Sparkplug-compliant MQTT server will broker pub/sub messages, but will not store and process NBIRTH and NDEATH messages. In contrast, a Sparkplug-aware MQTT server fully supports the Sparkplug B 3.0.0 standard. Refer to Chapter 10 in the Sparkplug B 3.0.0 Specification for details. [6]

Vendor support for Sparkplug B is still emerging. Some brokers are Sparkplug-aware, whereas some are only Sparkplug-compliant. Organizations must deploy a Sparkplug-aware broker to realize the benefits of continuous-state awareness and RBE. Sample MQTT brokers with Sparkplug B support include AWS IoT Core, Azure IoT Hub, Cirrus Chariot, EMQX, HiveMQ, Rabbit MQ and Solace Event Broker.

Unified Namespace

Many organizations use the ISA-95 Part 2 equipment model hierarchy to define a structure for their Sparkplug B group ID (refer to Figure 11). This structure provides a common naming convention for OT data. Some technical professionals use the term Unified Namespace to refer to this semantic name hierarchy and the underlying technology stack (e.g., MQTT, Sparkplug B) that makes it possible to discover data within this name hierarchy.

Figure 11: Unified Namespace


Unified Namespace
spBv1.0/group_id/message_type/edge_node_id/[device_id]
• Enterpise
• Site
• Area
• Line
• Cell
ISA-95 Equipment Model Hierarchy

Source: Gartner
792965_C

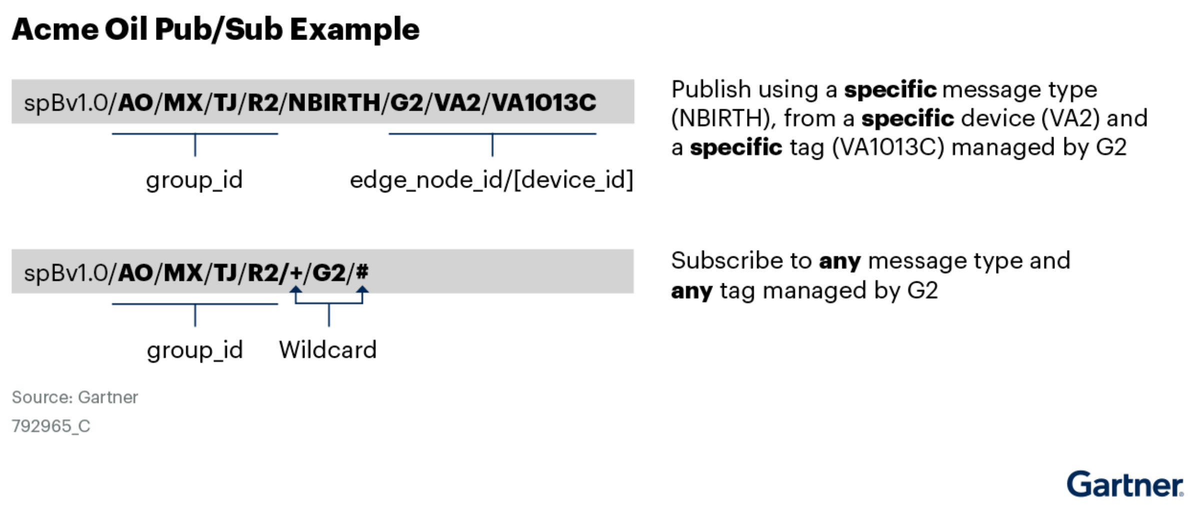
For example, consider the fictitious Acme Corporation. Acme is a multinational company with three divisions: Acme Oil, Acme Gas, Acme Solar. In this example, we focus on the Acme Oil division where they have three refineries: R1 (in Ontario), R2 (Tijuana), R3 (Louisiana). The organization defined the identifiers listed below for their company codes, country codes, site codes and so on. (Bolded terms are those appearing in Figure 12.)

- Company (**AO**, AG, AS): Acme Oil, Acme Gas, Acme Solar
- Country (US, CA, **MX**): USA, Canada, Mexico
- Site (LO, ON, **TJ**): Louisiana, Ontario, Tijuana
- Area (R1, **R2**, R3, LNG1/2, SA1/2): Refineries 1, 2 and 3, Liquid Natural Gas Plants 1 and 2, Solar Arrays 1 and 2
- Device (G1, **G2**, S1, S2): Gateways 1 and 2, Servers 1 and 2
- System (VA1, **VA2**, TA1, TA2, PA1, PA2): Valves 1 and 2, Tanks 1 and 2, Panels 1 and 2

- Equipment tag: **VA1013C**, TA2031A, PA5531

Figure 12 illustrates their topic structure for the bold codes to identify a specific tag (VA1013C) on a valve (V1) attached to a device (G2).

Figure 12: Acme Oil Pub/Sub Example


Acme Oil Pub/Sub Example
spBv1.0/**AO**/**MX**/**TJ**/**R2**/**NBIRTH**/**G2**/**VA2**/**VA1013C**
group_id
edge_node_id/[device_id]
spBv1.0/**AO**/**MX**/**TJ**/**R2**/**+**/**G2**/**#**
group_id
Wildcard
Gartner

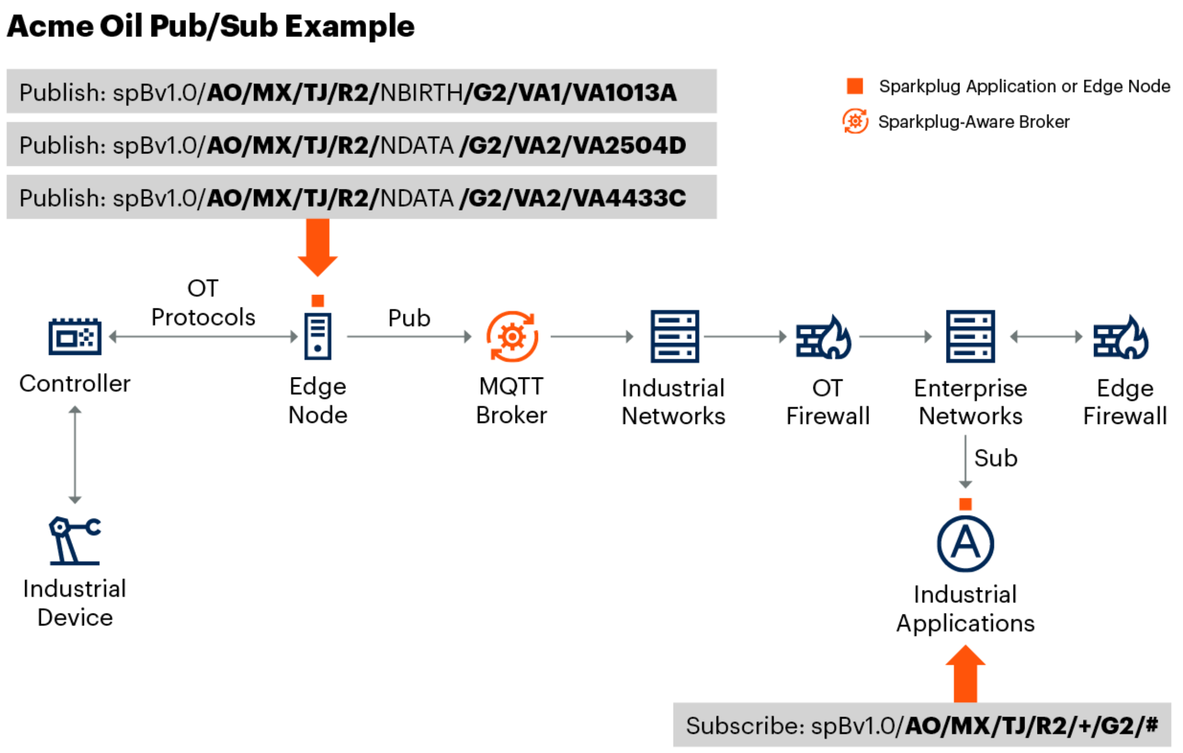
Publish using a **specific** message type (NBIRTH), from a **specific** device (VA2) and a **specific** tag (VA1013C) managed by G2

Subscribe to **any** message type and **any** tag managed by G2

Source: Gartner
792965_C

Figure 13 illustrates the publish and subscribe topics with the reference architecture. The industrial application subscribes to end-node G2 messages with wildcards + and #. When edge node G2 sends an NBIRTH message, the application automatically discovers *all* the metrics, and their current states from G2. The application receives all three of the messages published by G2. If the organization adds new sensors and tags to G2, then G2 will publish a new NBIRTH message. The industrial application receives this NBIRTH message, thus enabling it to dynamically discover the new sensor/tag metrics, as well as updated values for the previously reported sensors/tags.

Figure 13: Acme Oil Pub/Sub Example


Acme Oil Pub/Sub Example
Publish: spBv1.0/**AO/MX/TJ/R2/**NBIRTH**/G2/VA1/VA1013A**
Publish: spBv1.0/**AO/MX/TJ/R2/**NDATA **/G2/VA2/VA2504D**
Publish: spBv1.0/**AO/MX/TJ/R2/**NDATA **/G2/VA2/VA4433C**
Sparkplug Application or Edge Node
Sparkplug-Aware Broker
Controller
OT Protocols
Edge Node
Pub
MQTT Broker
Industrial Networks
OT Firewall
Enterprise Networks
Edge Firewall
Sub
Industrial Device
Industrial Applications
Subscribe: spBv1.0/**AO/MX/TJ/R2/+/G2/#**

Source: Gartner

MQTT = MQ Telemetry Transport; OT = Operational Technology

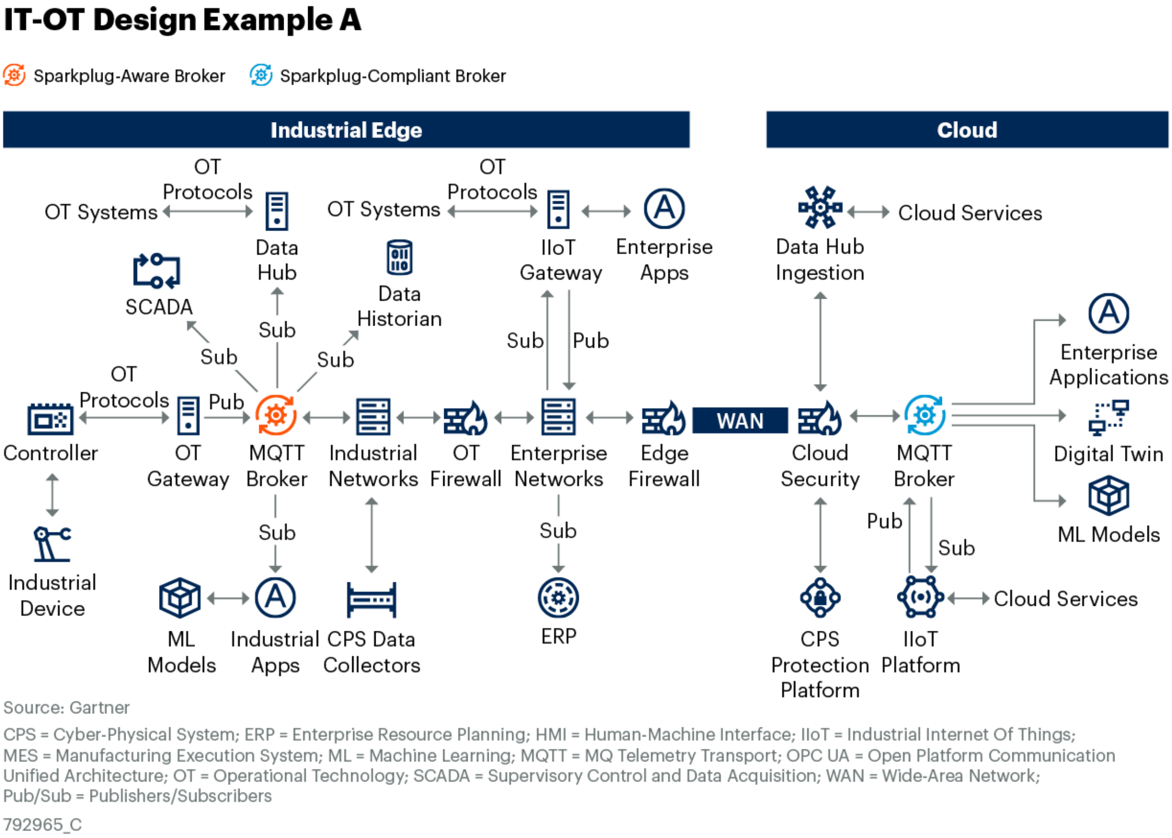
792965_C

IT-OT Design Examples

There are many ways to interconnect the architecture components. This section will describe two illustrative design examples.

IT-OT design example A (Figure 14) provides an event-centric architecture with a Sparkplug-aware broker, but without Sparkplug applications or edge nodes. Some organizations may choose to use this design as a transitional step toward supporting Sparkplug. This design will take several years to deploy, as many devices do not yet support MQTT. Even if equipment supports MQTT, it will take time for technical professionals to install, test and deploy software in all equipment at all industrial locations. The following provides a brief description of each of the components in this design example.

Figure 14: IT-OT Design Example A



Source: Gartner

CPS = Cyber-Physical System; ERP = Enterprise Resource Planning; HMI = Human-Machine Interface; IIoT = Industrial Internet Of Things; MES = Manufacturing Execution System; ML = Machine Learning; MQTT = MQ Telemetry Transport; OPC UA = Open Platform Communication Unified Architecture; OT = Operational Technology; SCADA = Supervisory Control and Data Acquisition; WAN = Wide-Area Network; Pub/Sub = Publishers/Subscribers

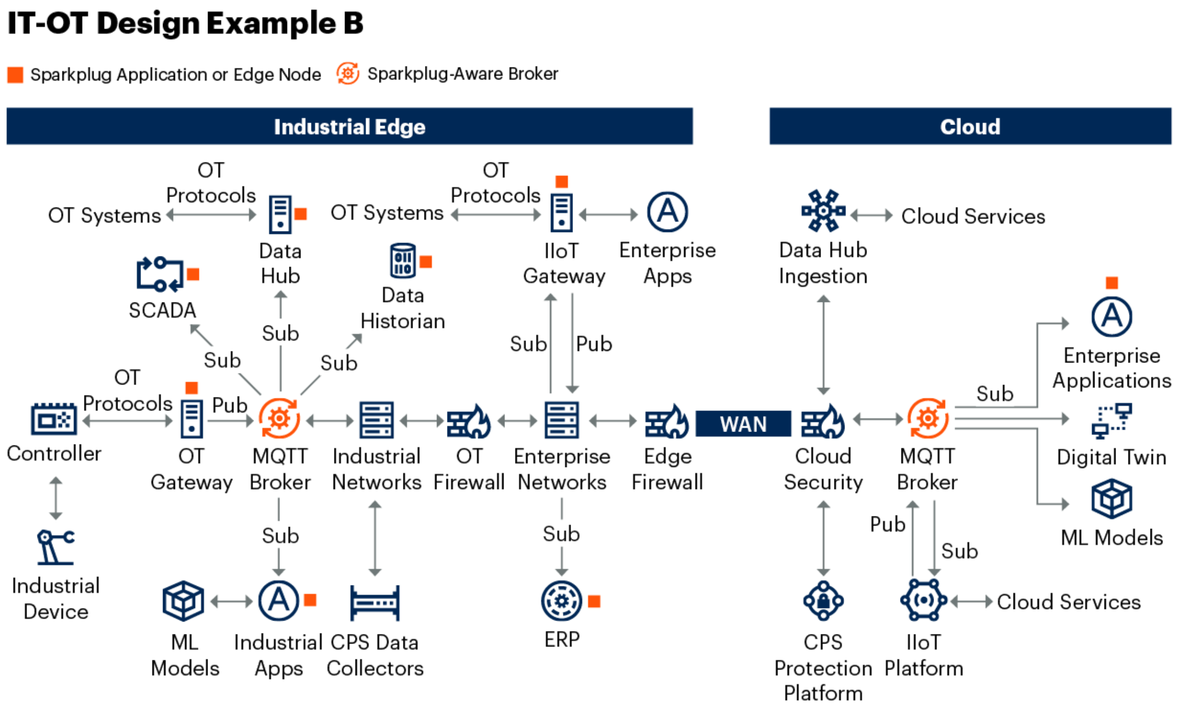
- **Edge MQTT Broker:** The edge Sparkplug-aware MQTT broker can process Sparkplug and non-Sparkplug messages. Organizations should install a broker that is compliant with MQTT 5.0 and Sparkplug 3.0.

- **Cloud MQTT Broker:** The cloud Sparkplug-compliant MQTT broker cannot participate in the Sparkplug protocol (e.g., it does not have the ability to store NBIRTH and DBIRTH messages). But it can forward Sparkplug messages from publisher to subscriber.

- **Broker-to-Broker Bridging:** Some brokers support the ability for one broker (e.g., the edge broker) to “bridge” pub/sub messages to another broker (e.g., the cloud broker). This enables an edge device to publish a message to an edge broker, and for a cloud application to subscribe to the message through a cloud broker. But broker to broker bridging is a proprietary, vendor-specific feature; thus, it will be difficult to change to a different broker in the future. Alternatives include:
  - Publication of applications directly to the edge and/or cloud brokers as needed.
  - Consumption of MQTT messages by the data hub, followed by the use of another technology (e.g., file transfer, Kafka) to transfer data from the edge to the cloud.
- **Edge Node:** OT equipment that does not support MQTT will use their OT protocols to communicate with an edge node. The edge node translates the OT protocol to MQTT.
- **Data Hub:** The hub consumes data from the MQTT broker as an MQTT subscriber, and from OT systems using OT protocols. It transfers that data to the cloud using various methods (Kafka streams, file transfer, database synchronization, etc.).
- **Applications:** Industrial applications (e.g., SCADA) and enterprise applications (e.g., ERP) consume MQTT events by subscribing to predefined topics. But they do not have the ability to consume Sparkplug messages.
- **Industrial IoT:** The IIoT gateway consumes messages from OT equipment using OT protocols. It may also perform edge processing (e.g., stream analytics). The gateway communicates with the IIoT platform by sending event messages to the cloud MQTT broker. Most IIoT platforms support the MQTT protocol.
- **CPS Protection Platform:** The CPS data collectors interoperate with the CPS protection platform to discover and protect cyber-physical assets. Organizations may deploy the CPS protection platform in the cloud or on-premises.

IT-OT design example B (Figure 15) provides an event-centric architecture with Sparkplug-aware brokers, and Sparkplug applications and edge nodes. The design provides the ability to consume Sparkplug messages at the edge and in the cloud, and thus realize the benefits of Sparkplug (e.g., RBE, data discovery, continuous state awareness). This design may take several years to deploy, because many devices do not yet support Sparkplug. Even if equipment supports Sparkplug, it will take time for technical professionals to install, test, and deploy software in all equipment at all industrial locations. The following provides a brief description of each of the components in this design example.

Figure 15: IT-OT Design Example B



Source: Gartner

CPS = Cyber-Physical System; ERP = Enterprise Resource Planning; HMI = Human-Machine Interface; IIoT = Industrial Internet Of Things; MES = Manufacturing Execution System; MQTT = MQ Telemetry Transport; OPC UA = Open Platform Communication Unified Architecture; OT = Operational Technology; SCADA = Supervisory Control and Data Acquisition; WAN = Wide-Area Network; Pub/Sub = Publishers/Subscribers

792965_C

- **Edge and Cloud MQTT Broker:** The edge and cloud Sparkplug-aware MQTT brokers can process Sparkplug and non-Sparkplug messages. Organizations should install a broker that is compliant with MQTT 5.0 and Sparkplug 3.0.

- **Broker-To-Broker Bridging:** See the analysis for design example A.

- **Edge Node:** OT equipment that does not support MQTT or Sparkplug will use their OT protocols to communicate with an edge node. The edge node translates the OT protocol to MQTT and Sparkplug.

- **Data Hub:** The hub consumes data from the MQTT broker using MQTT/Sparkplug, and from OT equipment using OT protocols. It transfers that data to the cloud using various methods (Kafka streams, file transfer, database synchronization, etc.).

- **Applications:** Industrial applications (e.g., SCADA) and enterprise applications (e.g., ERP) consume MQTT/Sparkplug events by subscribing to predefined topics. Applications are Sparkplug-aware, and thus can process the Sparkplug payload.

- **Industrial IoT:** The IIoT gateway consumes messages from OT equipment using OT protocols. It may also perform edge processing (e.g., stream analytics). The gateway communicates with the IIoT platform by sending event messages to the cloud MQTT broker.

- **CPS Protection Platform:** The CPS data collectors interoperate with the CPS protection platform to discover and protect assets. Organizations may deploy the CPS protection platform in the cloud or on-premises.

Strengths

This reference architecture has the following strengths:

- Provides an event-centric architecture that decouples producers and consumers in time, space and synchronization. The decoupling enables the addition and removal of producers and consumers without impacting the rest of the system, thus improving scalability.

- Makes use of open-source standards MQTT and Sparkplug B, which define message formats and publisher/subscriber behavior. Uses a widely supported data serialization format (Google protocol buffers). These features improve interoperability and vendor independence.

- Provides continuous-state awareness and RBE, thus eliminating the need for applications to poll devices to determine session state and current tag values.

- Provides a semantic name hierarchy, enabling applications to autodiscover new data tags and their current values.

- Integrates a data hub that improves operational effectiveness through consistent governance of data and analytics across sets of diverse endpoints. Also reduces cost and complexity, compared with the Purdue Model's point-to-point integration design.
- Integrates CPS protection platforms that provide asset discovery, visibility, protection and network topology. The platforms integrate with SIEM and SOAR solutions to provide unified IT-OT security.
- Integrates an IIoT platform that collects high-volume time series and/or low-latency complex machine data from networked IoT endpoints. The IIoT platform also orchestrates historically siloed data sources to enable better accessibility, and improve insights and actions across a heterogeneous asset group.
- Facilitates scoping, governance and monitoring of digitization initiatives. Improves communications with vendors and between different IT and business stakeholders.

Weaknesses

This architecture has the following weaknesses:

- The reference architecture has not been widely adopted. There may be unforeseen issues as industrial organizations perform large-scale deployments.
- The Universal Namespace naming convention is dependent on knowing what assets you have. CPS-PP systems may not discover all OT assets. Thus, the UNS hierarchy may be incomplete.
- UNS requires organizations to define a unified naming hierarchy that the entire organization uses to define, access and manage their industrial business data. For many organizations, this may be a long and possibly contentious process, as it requires collaboration across many different roles and teams.
- MQTT and Sparkplug B are optimized for telemetry data publishing (one publisher to many subscribers). However, they do not support transaction exchanges (e.g., command/response exchange to load a new process manufacturing recipe into a machine). MQTT and Sparkplug B do not indicate that a subscriber received the message or that a subscriber acted upon the message. (Setting QoS=2 only indicates that the broker received the message from the publisher.) **Hence, organizations must use a mechanism other than MQTT and Sparkplug B for command/response exchanges.**

- Sparkplug B is optimized for tightly packing (i.e., binary encoding) all the data managed by a device (i.e., NDATA message) and sending that data to a subscriber application (e.g., SCADA system). But applications may want to request a single tag value, not all tag values. Sparkplug B is not able to support Query/Response exchanges for specific data/tags. **Hence, organizations will need to store data received from brokers into databases and data lakes to support application data queries.**

- The [group ID + edge node ID] couplet must be unique within the global namespace. A message containing a [group ID + edge node ID] must only identify a single broker destination. This may become a problem as different parts of the organization independently define their group and edge node IDs, and inadvertently use duplicate IDs. One solution is to define a pool of globally unique group IDs that are assigned to various teams and independently managed by those teams. Each team can then independently define their edge node IDs. This is like creating a pool of organizational unique identifiers for the first three bytes of a MAC address.

- Some people may perceive this reference architecture as too technical and lacking alignment with business processes. Thus, technical professionals may not be able to deploy the architecture.

Guidance

I&O technical professionals should take the following actions:

Establish an Industrial Center of Excellence

Establish an ICOE with an executive mandate and well-defined role. Appoint an enterprise architect as the organization’s chief industrial modernization architect to lead the ICOE. Drive transformation by ensuring the ICOE works collaboratively with industrial modernization stakeholders across the organization, including OT staff and IT staff. The ideal ICOE is a business-outcome-driven enterprise architecture function that provides governance (e.g., management of universal name service) and leads the transformation activities that are necessary for a successful industrial modernization journey.

Obtain CIO Mandate

CIOs have increasing responsibility for OT management (see Survey Analysis: IT/OT Alignment and Integration). It is important for an ICOE to have CIO sponsorship, because it will be setting policies that need to be respected by the organization. There must be widespread recognition of the ICOE's authority to set industrial modernization policy and cooperation from other teams that will enforce industrial modernization policies. There needs to be awareness of the ICOE's existence and its ability to consult on IT-OT projects. Thus, it needs to be "marketed" internally by its executive sponsor.

Appoint ICOE Lead Architect

The ICOE architect should be led by a well-respected long-term employee of the organization. The role of ICOE leader is a major responsibility that may carry a formal job title. The leader should have the following skills: change leadership, thorough business knowledge, operational technology proficiency, information technology proficiency, security proficiency, enterprise architect skills and data analytics knowledge. The last skill is particularly important, because managing the data pipeline is a central issue for the industrial modernization journey.

Provide Architectural Guidance

Architectural guidance includes creating ICOE reference architectures, setting standards (e.g., a semantic name hierarchy) and creating an approved vendor list (e.g., MQTT brokers). Establish and maintain a shared repository of these artifacts. Establish a process for review, signoff and governance of reference architectures and standards. Create and deliver training sessions to proliferate understanding, acceptance of and adherence to reference architectures and standards.

Create an Industrial Modernization Technical Strategy Document

Describe your purpose and vision. Describe the business problems you are trying to solve and the desired measurable outcomes. Describe the multidomain technology components (e.g., OT, infrastructure; security, data and analytics; and software development).

Use an Agile Approach

Agile is as much a mindset as a methodology. Its core tenets include collaboration, empowerment, transparency and continuous improvement. It upholds these by focusing on frequent feedback that enables a team to respond to change. The agile approach leans heavily on the creativity and collaboration of participants.

Monitor Changes in Standards and Industry Practices

The ICOE must stay abreast of changes in the industry. For instance, the Weaknesses section of this document highlighted specific problems that this architecture does not solve (e.g., inability to handle command/response use cases). The Sparkplug working group is aware of these issues and is working on version 4.0. Vendors and industrial organizations may implement these changes. The ICOE must evaluate how and when to deploy these changes.

Evidence

This research used the anonymized information drawn from Gartner client inquiry and from IT-OT technical documents shared by Gartner clients.

[1] The Industrial Internet Reference Architecture, Industry IoT Consortium.

[2] Reference Architectural Model Industrie 4.0, VDI/VDE Society Measurement and Automatic Control (GMA).

[3] OASIS Message Queueing Telemetry Transport (MQTT) TC, OASIS Open.

[4] The NIS2 Directive: A High Common Level of Cybersecurity in the EU, European Parliament.

[5] Efficient IIoT Communications: A Comparison of MQTT, OPC-UA, HTTP, and Modbus, Cirrus Link.

[6] Sparkplug Version 3.0, Sparkplug.

Recommended by the Author

Some documents may not be available as part of your current Gartner subscription.

Streamline Integration by Choosing Between Data-, Event- or Application-Centric

Styles Choosing Event Brokers: The Foundation of Your Event-Driven Architecture

How to Operationalize Data Workloads

Data and Analytics Essentials: DataOps

Data and Analytics Essentials: Data Hubs

Implementing the Technical Architecture for Master Data Management

CPS Security Governance — Best Practices From the Front Lines

Market Guide for Operational Technology Security

Magic Quadrant for Global Industrial IoT Platforms

Architect IoT Using the Gartner Reference Model

Actionable, objective insight

Position your IT organization for success. Explore these additional complimentary resources and tools for I&O and IT leaders:

Roadmap

Craft a Cloud Strategy to Optimize Value

Maximize the benefits of the cloud with a strategy that clarifies the cloud's role in delivering IT-driven business value.

Download Now

Webinar

Develop the Infrastructure Workforce of the Future

Overcome skills shortages through a culture of development in infrastructure organizations.

Watch Now

Resource Hub

Gartner for Infrastructure & IT Operations Leaders

Explore forward-thinking insights that help heads of infrastructure and IT operations reinvent how they deliver value for the business.

Read Now

Tool

Heads of Infrastructure & IT Operations Effectiveness Diagnostic

Benchmark your performance and develop key actions and behaviors that differentiate highly effective I&O and IT leaders.

Learn More

Already a client?
Get access to even more resources in your client portal. Log In

Connect With Us

Get actionable, objective insight that drives smarter decisions and stronger performance on your mission-critical priorities. Contact us to become a client:

U.S.: 1 855 811 7593

International: +44 (0) 3330 607 044

Become a Client

Learn more about Gartner for IT Leaders
gartner.com/en/information-technology

Stay connected to the latest insights

Attend a Gartner conference
View Conference

Gartner