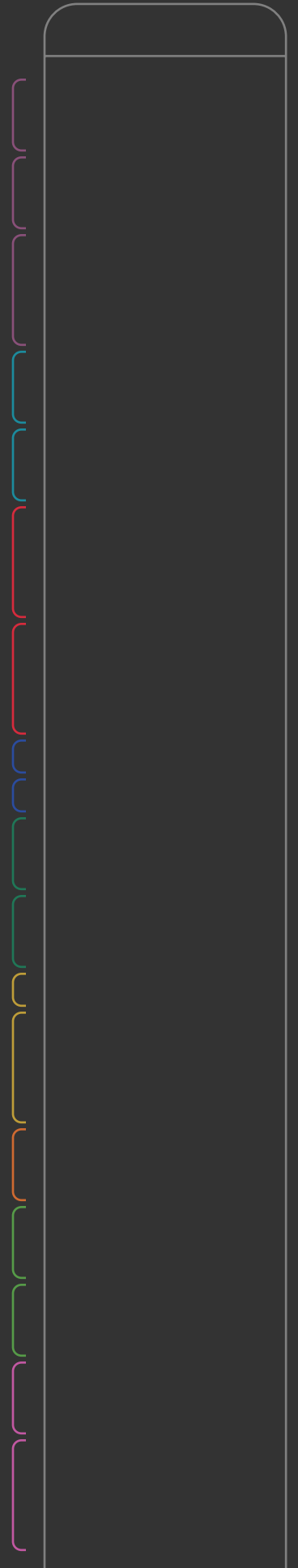


MIT-E Results

December 18, 2017

MachNation



What is MIT-E?

MIT-E is an IoT platform test lab run by MachNation, the world leading analyst firm researching IoT platforms and middleware. MIT-E's core product is a database to help enterprises compare IoT platform usage statistics. MIT-E analysts perform a set of common hands-on tasks – developer workflows – on IoT platforms. MIT-E analysts score these tasks and compile them in a database. MIT-E scores platform tasks based on the time-to-complete each task, ease-of-completion, completeness of tasks and sophistication metrics. MachNation makes the MIT-E database and detailed reports available to enterprises to help guide IoT platform purchase decisions. The database and detailed reports provide enterprises an apples-to-apples comparison of IoT platforms across relevant hands-on task metrics.

Legal Disclaimer

MachNation grants purchaser a single user-license for use of this report. All contents of this report and all other MachNation IoT Test Environment (MIT-E) reports are and remain the intellectual property of MachNation. This report and all MIT-E reports may not be posted publicly in hard or soft copy formats by purchaser. Purchaser may not post this report or any other MIT-E report on any publicly accessible website. Purchaser may not publicly use or disclose any of the contents in this report or any other MIT-E report without the written permission of MachNation, such permission to be granted or denied at MachNation's sole discretion. Purchaser may not resell this report or any other MIT-E report.

Methodology

C Completeness of Task

Measure of how completely the evaluated product executed the task requirements as defined in the task description.

SCORE RANGE	0 to 3
SCORE 0	Not completed or functionality not available in product (<50% completion)
SCORE 1	Partially completed task (50% to 74% completion)
SCORE 2	Mostly completed task (75% to 99% completion)
SCORE 3	Fully completed task (100% completion)

S Sophistication of Solution

Measure of how effectively and with what level of sophistication the evaluated product executed the requirements as defined in the task description.

SCORE RANGE	0 to 3
SCORE 0	Very unsophisticated solution with regard to task execution (e.g., unclear documentation, poor UI, bad design)
SCORE 1	Somewhat unsophisticated solution with regard to task execution (e.g., unclear documentation, UI, design)
SCORE 2	Somewhat sophisticated solution with regard to task execution (e.g., good documentation, UI, design)
SCORE 3	Very sophisticated solution with regard to task execution (e.g., excellent documentation, UI, design)

E Ease of Task Completion

Calculated statistic describing the relative percentile of a single Timing of Task measure for a single given task, relative to the aggregate Timing of Task measures for all tested products and vendors for the same given Task.

SCORE RANGE	0 to 3
SCORE 0	Bottom 25% (Slowest) of all Timing of Task measures for the given Task
SCORE 1	Bottom 50% (Slow) of all Timing of Task measure for the given Task
SCORE 2	Top 50% (Fast) of all Timing of Task measures for the given Task
SCORE 3	Top 25% (Fastest) of all Timing of Task measures for the given Task

T Timing of Task

Measure of how long the task execution took to complete in minutes. Timing values only assigned to tasks that were "fully completed" as per Completeness of Task criteria. Does not include timing of: initial familiarization with product, research of items/documentation/elements not directly related to the task description, preliminary configuration/ setup not directly related to task description, or time spent waiting on "blocking" elements outside of core task description requirements.

SCORE RANGE	0 to infinite
TIMING	Total time to execute task in minutes
DNF	Task could not be completed

Amazon

Product Name

Product Version

AWS IoT

2017.08.24

AWS IoT Summary

ACCESS CONTROL

COMPLETENESS OF TASK



SOPHISTICATION OF SOLUTION



EASE OF TASK COMPLETION



DEVICE MANAGEMENT

COMPLETENESS OF TASK



SOPHISTICATION OF SOLUTION



EASE OF TASK COMPLETION



DATA MANAGEMENT

COMPLETENESS OF TASK



SOPHISTICATION OF SOLUTION



EASE OF TASK COMPLETION



ANALYTICS

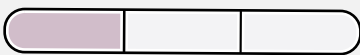
COMPLETENESS OF TASK



SOPHISTICATION OF SOLUTION



EASE OF TASK COMPLETION



EVENT PROCESSING

COMPLETENESS OF TASK



SOPHISTICATION OF SOLUTION



EASE OF TASK COMPLETION

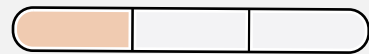


EXTERNAL INTEGRATION

COMPLETENESS OF TASK



SOPHISTICATION OF SOLUTION



EASE OF TASK COMPLETION



MONITORING

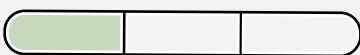
COMPLETENESS OF TASK



SOPHISTICATION OF SOLUTION



EASE OF TASK COMPLETION



USABILITY

COMPLETENESS OF TASK



SOPHISTICATION OF SOLUTION



COMPREHENSIVE EVAL

COMPLETENESS OF TASK



SOPHISTICATION OF SOLUTION



Access Control

SUMMARY

Amazon provides a comprehensive framework for both device and user authentication. AWS IoT supports both certificate- and credential-based authentication for users as well as certificate-based authentication for devices. By extending the existing AWS access control framework including AWS Identity and Access Management (IAM) and AWS Cognito to the AWS IoT service, administrators and operators are able to leverage Amazon's best-in-class authentication and access management in the IoT domain. In particular, AWS IAM allows for extremely fine-grained access control both in terms of platform configuration and device and data access. Finally, Amazon provides a number of tools to aid in the provisioning and management of X.509 certificates, largely ameliorating the need for third-party certificate or PKI management tools.

C = COMPLETENESS OF TASK

E = EASE OF TASK COMPLETION

S = SOPHISTICATION OF SOLUTION

T = TIMING OF TASK

ACCESS CONTROL			C	S	E	T
DEVICE AUTHORIZATION	01-01-01	Configure user multi-factor authentication	3	2	2	3
	01-01-02	Create a new user	3	2	3	2
MULTI-TENANT CAPABILITIES	01-01-03	Create bulk new users	3	2	2	2
	01-02-01	Authorize bulk device credentials	3	2	3	2
USER AUTHORIZATION	01-02-02	Authorize device credentials	3	2	3	1
	01-03-01	Create subordinate user or organization in hierarchy	3	2	3	1
	01-03-02	Restrict permissions of subordinate user or organization	3	2	2	3

Device Management

SUMMARY

Device management is perhaps the single most lacking component of Amazon's IoT platform. Until the recent release of Amazon's AWS Greengrass, AWS IoT was entirely devoid of device management functionality. AWS Greengrass provides a number of new capabilities including limited device and gateway management and managed edge computing functionality. AWS Greengrass does not provide many of the typical IoT lifecycle management functionalities such as firmware management or device diagnostics and sophisticated status reporting. AWS Greengrass does provide several compelling gateway and edge computing capabilities such as enabling AWS Lambda functions to be executed at the edge or allowing managed gateways to gracefully handle intermittent device-to-cloud communication. While AWS Greengrass is a good first-step for Amazon, it remains to be seen if this strategy will continue to evolve into a fully-fledged device management offering.

C = COMPLETENESS OF TASK

E = EASE OF TASK COMPLETION

S = SOPHISTICATION OF SOLUTION

T = TIMING OF TASK

DEVICE MANAGEMENT		C	S	E	T
DEPLOYMENT	02-01-01 Add new managed logical device to platform	3	2	1	7
	02-01-02 Add new unmanaged logical device to platform	3	3	2	3
	02-01-03 Configure/compile device agent for a device	3	2	3	9
	02-01-04 Deploy device agent and establish bidirectional communication	3	2	1	14
	02-01-05 Deploy device and establish unidirectional communication	3	2	3	2
CONNECTIVITY	02-02-01 Monitor current network status	3	1	3	1
	02-02-02 Monitor historical network status	0	0	DnF	DnF
DEVICE DEFINITIONS/ GROUP	02-03-01 Assign single device to a group	3	2	3	2
	02-03-02 Configure customer-defined metadata parameter	3	3	3	2
	02-03-03 Configure device metadata parameter	3	3	3	1

C - COMPLETENESS OF TASK

E - EASE OF TASK COMPLETION

S - SOPHISTICATION OF SOLUTION

T - TIMING OF TASK

DEVICE MANAGEMENT, CONT.			C	S	E	T
MONITORING	02-04-01	View device status for a single device	3	2	3	1
	02-04-02	View device status for multiple devices	3	1	2	5
	02-04-03	View historical device status of a single device	3	2	2	9
EDGE MONITORING	02-05-01	View sensor data for multiple downstream devices	3	3	3	2
	02-05-02	View sensor data for a downstream device	3	3	3	2
EDGE MANAGEMENT	02-06-01	Push a configuration to a downstream device	0	0	DnF	DnF
FIRMWARE/SOFTWARE	02-07-01	Push firmware to a single device	0	0	DnF	DnF
	02-07-02	Push software to a single device	0	0	DnF	DnF
	02-07-03	Update/create firmware image on the platform repository	0	0	DnF	DnF
REMOTE CONFIG/ACTIONS	02-08-01	Remotely trigger command for a single device	2	3	DnF	DnF
	02-08-02	Remotely update device parameter for a single device	3	3	1	4
SUNSETTING	02-09-01	Remove a single device from the platform	3	3	3	1
DIAGNOSTICS/TROUBLESHOOTING	02-10-01	Obtain audit and/or config logs for the platform	3	2	3	2
	02-10-02	Obtain diagnostic log for a single device	3	2	3	1
	02-10-03	Reboot remote device	0	0	DnF	DnF

Data Management

SUMMARY

Data Management is an area of distinct strength for AWS IoT. Independent of AWS IoT, Amazon provides an extremely comprehensive suite of data management and data storage services including managed SQL services, cloud-based NoSQL services and extensive options for data warehousing and data analytics. However, both AWS IoT and the broader Amazon ecosystem are somewhat deficient in terms of operator usability in terms of data management. Like with Amazon's analytics services, many data services lack pre-configured operator interfaces or dashboards, instead relying on developer/operator competency with SQL-style syntax. This can make casual visibility into stored data a less than smooth experience, although for customers seeking capability and extensibility at the expense of usability, this may pose less of an impediment.

C = COMPLETENESS OF TASK

E = EASE OF TASK COMPLETION

S = SOPHISTICATION OF SOLUTION

T = TIMING OF TASK

DATA MANAGEMENT				C	S	E	T
INGRESS	}	03-01-01	Configure persistent on-platform data storage	3	3	1	14
EGRESS		03-02-01	Forward live sensor data to external endpoint	2	2	DnF	DnF
RETRIEVAL	}	03-03-01	View historical sensor data from a group of devices	3	2	1	31
		03-03-02	View historical sensor data from a single device	3	2	1	19
MANAGEMENT	}	03-04-01	Compute aggregate statistics for multiple data points	3	3	1	26
		03-04-02	Compute aggregate stats for single data point	3	3	1	25
		03-04-03	Delete a single historical sensor data point	3	2	2	4

Analytics

SUMMARY

Amazon's current IoT analytics implementation is flexible and extensible, but lacking in overall usability. For on-platform IoT data analytics, Amazon provides the Kinesis Analytics service that ingests data from Kinesis Firehouse and Kinesis Streams (fed by AWS IoT) and is capable of executing streaming analytics on petabyte-scale data. While Amazon has provided some pre-built integration with the open-source Kibana application, Kinesis Analytics is currently lacking operator interfaces. While its potent capability is sure to please data scientists, less sophisticated operators or those looking for a casual data discovery platform should consider more user-friendly alternatives.

C = COMPLETENESS OF TASK

E = EASE OF TASK COMPLETION

S = SOPHISTICATION OF SOLUTION

T = TIMING OF TASK

ANALYTICS			C	S	E	T
ANALYTICS BASIC	04-01-01	Build an on-platform analytics report	3	2	2	7
	04-01-02	Configure external analytics service for live streaming data	0	1	DnF	DnF
	04-01-03	Configure on-platform analytics service for live streaming data	3	3	2	26

Event Processing

SUMMARY

Amazon provides multiple effective way for achieving complex event processing with the AWS IoT service. The most straightforward method is via the AWS IoT Rules service. The AWS IoT Rules service provides a flexible option for routing data from initial data ingestion to the various other AWS services. The AWS IoT Rules are defined using a SQL-like syntax allowing operators/administrators to filter incoming data streams in real-time and either trigger actions or route data based on matching conditions. In addition, more complex event processing including advanced statistical analyses and data transformations are made available via AWS Kinesis Streams, AWS Kinesis Analytics and AWS Kinesis Firehouse. AWS IoT rules provides a more user-friendly and mostly-declarative process for rule implementation while the various AWS Kinesis services offer options for those customers with more complex requirements which may exceed the functionality of IoT Rules.

C = COMPLETENESS OF TASK

E = EASE OF TASK COMPLETION

S = SOPHISTICATION OF SOLUTION

T = TIMING OF TASK

EVENT PROCESSING			C	S	E	T
PLATFORM RULES ENGINE / CEP BASIC	05-01-01	Configure rule/alert for sensor data configured range exceeded	3	3	3	4
	05-01-02	Configure rule/alert on device status	3	1	2	16
PLATFORM RULES ENGINE / CEP ADVANCED	05-02-01	Configure rule/alert for sensor data based on anomaly detection	3	1	2	28
	05-02-02	Configure rule/alert for sensor data (real-time moving average)	3	2	3	9
EDGE RULES ENGINE / CEP	05-03-01	Configure edge rule/alert for device status	3	2	2	17
	05-03-02	Configure edge rule/alert for sensor data range exceeded	3	3	3	4

External Integration

SUMMARY

External integration is an area of weakness for Amazon. While AWS IoT is heavily integrated with the other services provided through the AWS cloud, it severely lacks integration to external services and external providers. Though AWS IoT rules provide a method of integrating IoT data streams into Salesforce's nascent IoT Hub service, this Salesforce service is not yet generally available. As the sole third-party integration provided, this option is extremely limited for Amazon's customers. In addition, forwarding of incoming MQTT/HTTP IoT data streams to off-platform ingestion points is not supported out-of-the-box, requiring custom code implementation via an AWS Lambda function to stream data out of the platform.

C = COMPLETENESS OF TASK

E = EASE OF TASK COMPLETION

S = SOPHISTICATION OF SOLUTION

T = TIMING OF TASK

EXTERNAL INTEGRATION				C	S	E	T
CRM/ERP	{	06-01-01	Configure rule/platform for outbound pre-integrated app action	3	1	3	5
		06-02-01	Configure credentials for external application	3	3	2	5
CUSTOM APPLICATIONS	{	06-02-02	Configure rule/platform for outbound external app action	3	1	3	3
		06-02-03	Trigger device command from external application request	0	0	DnF	DnF

Monitoring

SUMMARY

Monitoring of IoT assets and data streams is an area of mixed strength for AWS IoT. While operator and administrator interfaces are not completely lacking basic functionality, most are designed around the assumption of a limited number of devices being deployed while being more well-suited to test or trial deployments than large operational deployments. Visibility into device state/status is also relatively limited, though the newly released AWS Greengrass does provide some improvements for those devices running the platform's managed agent. In addition, limited bulk device views and the omission of any user editable dashboards or interfaces means customers will undoubtedly need to develop and build their own operational interfaces for all but the most simple IoT deployments.

C = COMPLETENESS OF TASK

E = EASE OF TASK COMPLETION

S = SOPHISTICATION OF SOLUTION

T = TIMING OF TASK

MONITORING			C	S	E	T
ALERTING	07-01-01	Trigger email on event	3	2	2	9
	07-01-02	Trigger external-call on event	2	2	DnF	DnF
	07-01-03	Trigger SMS on event	3	2	3	9
DASHBOARDS	07-02-01	Create a dashboard with filtered and aggregated sensor data	0	0	DnF	DnF
EDGE DASHBOARDS	07-03-01	Create an edge dashboard with filtered alert status	0	0	DnF	DnF
SINGLE VIEWS	07-04-01	View device alert status for a single device	3	2	2	5
	07-04-02	View sensor data for a single device	3	3	3	1
BULK VIEWS	07-05-01	View device alert status for bulk devices	3	1	2	8
	07-05-02	View sensor data for bulk devices	3	2	1	7

Usability

SUMMARY

Usability is overall quite well provided within AWS IoT. In particular, Amazon excels at developer usability with an excellent developer portal, clear and thorough documentation throughout the platform and very complete platform APIs. Device SDKs are a bit lacking compared to other platforms especially with the newly released Greengrass service. In addition, the device APIs are somewhat limited in terms of functionality provided out-of-the-box, though they are fairly extensible in practice. As with monitoring, the most deficient aspect of the Amazon platform is the lack of pre-built operator interfaces, requiring customers to instead rely on custom interface development. Overall, Amazon provides an excellent usability experience especially for those customers with the patience to dig into the provided feature set and build upon the foundational aspects provided by the platform.

C = COMPLETENESS OF TASK

E = EASE OF TASK COMPLETION

S = SOPHISTICATION OF SOLUTION

T = TIMING OF TASK

USABILITY		C	S	E	T	
DEVELOPER USABILITY	08-01-01	Locate and evaluate quality of developer portal	3	3	N/A	N/A
	08-01-02	Locate and evaluate quality of platform API	3	3	N/A	N/A
	08-01-03	Locate and evaluate quality of platform-level documentation	3	3	N/A	N/A
	08-01-04	Locate and evaluate quality of platform-provided device APIs	2	3	N/A	N/A
	08-01-05	Locate and evaluate quality of platform-provided device SDKs	2	2	N/A	N/A
OPERATOR/ ADMINISTRATOR USABILITY	08-02-01	Evaluate overall cogency/quality of administrator UI	3	2	N/A	N/A
	08-02-02	Evaluate overall cogency/quality of operator UI	1	1	N/A	N/A

Comprehensive Eval

SUMMARY

With a strong performance overall, Amazon's AWS IoT service is evolving into a serious competitor in the IoT space. With a class-leading security model and sufficient, yet not overwhelming scores in overall platform cogency and productization, AWS IoT is an excellent solution for a bare bones, greenfield IoT implementation. While the overall architecture model is satisfactory, it is focused mostly on leveraging existing AWS services into the IoT domain. Support for external products/services is quite limited and integration between AWS IoT and existing AWS services could be improved. However, for those customers with existing AWS deployments, AWS IoT is overall a worthy IoT solution.

C = COMPLETENESS OF TASK

E = EASE OF TASK COMPLETION

S = SOPHISTICATION OF SOLUTION

T = TIMING OF TASK

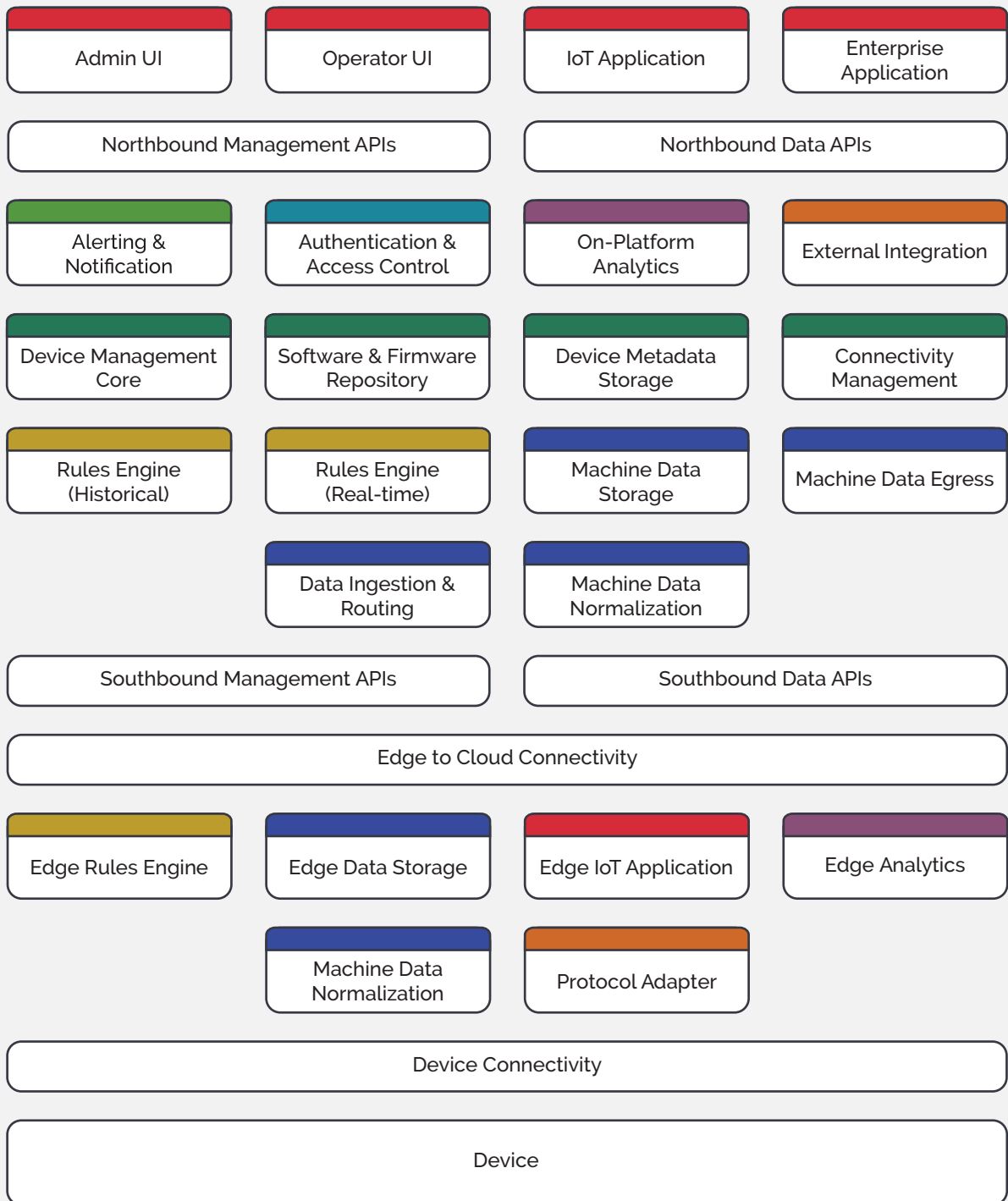
COMPREHENSIVE EVAL			C	S	E	T
ARCHITECTURE	09-01-01	Evaluate overall cogency and quality	2	2	N/A	N/A
	09-01-02	Evaluate overall security model and implementation	3	3	N/A	N/A
PRODUCTIZATION	09-02-01	Evaluate overall productization and integregation	2	2	N/A	N/A

MachNation IoT Architecture

CLOUD

EDGE

DEVICE



- APPLICATION
- MONITORING
- ACCESS CONTROL
- ANALYTICS
- INTEGRATION
- DEVICE MANAGEMENT
- EVENT PROCESSING
- DATA MANAGEMENT

Categories of IoT Platform Functionality

Administrator User Interface (UI)

The administrator user interface (UI) provides configuration management capabilities including access control and platform configuration. All capabilities are provided for the IoT platform and associated services.

Alerting and Notification

Alerting and notification is any system of pushing data, metadata and messages to operators, administrators or external systems for purposes of generating an action. Alerting and notifications may include UI and user experience (UX) elements in a dashboard, email or SMS message. Alerting and notifications might also use push-based or pull-based API/M2M elements to complete their message delivery purposes.

Authentication and Access Control

Authentication and access control is a system of identity verification and identity management for all platform-connected elements including APIs, administrator UI, operator UI, devices and platform-provided services. Authentication and access control should support multi-factor authentication for both users and devices. Authentication and access control may also include encryption and data protection though not required in all IoT cases.

Connectivity Management

Connectivity management refers to any system that orchestrates, configures, or otherwise manages the device connectivity layer. Such systems may manage provisioning, billing, or utilization metrics of the relevant wireless or wired technologies utilized by IoT devices or IoT gateways.

Data Ingestion and Routing

Data ingestion and routing is a service that allows platforms to ingest machine data from connected IoT devices, aggregation points and gateways. Data ingestion and routing is often a MQTT/HTTP endpoint, but is logically protocol agnostic. Data ingestion and routing relays collected data to rules engines, storage engines or external services.

Device

A device is a combination of hardware and software assembled to perform some IoT function. The hardware component is often comprised of an integrated circuit or system on chip (SoC), sensor, actuator, communication module and security module. The software component is often comprised of firmware and software packages, a boot loader, an operating system and a device agent.

Device Connectivity

Device connectivity is the communication service allowing data to travel from devices to an IoT edge gateway using Bluetooth low-energy, Zigbee/Z-Wave, LPWAN or other LAN/WAN technologies. Devices may also connect directly to the IoT platform without using an IoT gateway by using LPWAN, cellular, satellite, or fixed-line services. Typical communication protocols include MQTT or HTTP(S).

Device Management Core

Device management core is a service that provides a central repository and inventory of information for all connected or managed IoT devices, aggregation points and gateways. In addition the device management core exposes services that enable lifecycle management of devices.

Device Metadata Storage

Device metadata storage is an asset database that provides a collection point for all IoT device metadata including device current state and historical state. Very often device metadata storage is implemented as a SQL-type datastore. Device metadata storage can be exposed directly to the IoT platform or enterprise application, or can only be exposed internally to the IoT device management services.

Edge Analytics

Edge analytics is any type of data- and metadata-related quantitative exploration executed locally at the edge. Edge analytics often include limited anomaly detection or other basic security-related analytic services, though more complete analytic implementations are also possible.

Edge Data Normalization

Edge data normalization is a service that enables the conversion and standardization of machine data at the IoT edge from unstructured, streaming sources to compressed, structured data formats for northbound transmission or storage.

Edge Data Storage

Edge data storage is a service that provides either transient or long-term amassment of machine data at the IoT edge. Edge data storage can be used as a short-term storage engine during periods of intermittent platform connectivity or as a longer-term storage engine for edge-based analytics or monitoring.

Edge IoT Application

An edge IoT application is an IoT application deployed to and executed from the edge of an IoT network that typically interfaces with locally available resources and devices, but may also connect to southbound or northbound (data and management) APIs.

Edge Rules Engine

Edge rules engine or a complex event processing (CEP) engine is the ability to execute actions including external callouts, notifications and alerts executed on the edge of the IoT network. The edge rules engine is often a feature-limited version of the on-platform, cloud-based rules engine, though it may also be implemented as a fully-featured CEP.

Edge to Cloud Connectivity

Edge-to-cloud connectivity is the communication service allowing data to travel from IoT devices, aggregation points and gateways to cloud IoT platform and other cloud services. Connectivity options include low-power wide-area networks (LPWAN), cellular, satellite, proprietary networks and fixed-line services.

Enterprise Application

An enterprise application is any external service including a third-party analytics service, data-storage service and others, that interfaces with northbound (data and management) APIs to provide functionality to platform operators.

External Integration

An external integration is a solution using an API or other connector allowing bidirectional flow of data between an IoT platform and external systems or platforms including ERP, CRM/SFA, inventory management, trouble ticketing and others. External integrations, unlike generic machine data egress topologies, are productized offerings providing pre-built connectors to selected external systems or platforms. These external integrations allow the selective push of data based on business rules.

IoT Application

The IoT application interfaces with the northbound (data and management) APIs to provide access to platform data, non-platform data and configurations on the platform.

Machine Data Egress

Machine data egress is a service to programmatically provide data retrieval from either on-platform or off-platform data stores. Machine data egress usually allows users to create time series filters and queries against underlying NoSQL data stores.

Machine Data Normalization

Machine data normalization is a service that enables the conversion and standardization of machine data from unstructured, streaming sources to compressed, structured data formats for northbound transmission or storage.

Machine Data Storage

Machine data storage is a service that allows the amassment of IoT device data typically in time-series formats. Machine data storage provides services to allow querying of machine data based on IoT device or time period. It usually consists of a NoSQL data store, although relational data stores are also possible. Some IoT platforms provide no storage capabilities, some require usage of an external-to-platform data store and some provide limited periods of data retention.

Northbound Data APIs

Northbound data APIs are either a single API or collection of APIs facilitating management of data storage. The northbound data APIs provide programmatic access to data stored within the IoT platform as well as live data received from IoT devices.

Northbound Management APIs

Northbound management application programming interfaces (APIs) are either a single API or collection of APIs facilitating management of the configuration and operations of an IoT platform. The northbound management APIs may be separated into a device management API, operation API, administrator API and others.

On-Platform Analytics

On-platform analytics is any type of data- and metadata-related quantitative exploration executed in the cloud platform. On-platform analytics can include discrete analytics services, fully-integrated analytics services or vendor-provided applications.

Operator UI

The operator UI provides the day-to-day interface for platform operators for functions including device management, data management, reporting and analytics. All capabilities are provided for the platform and associated services.

Protocol Adapter

Protocol adapter is a service deployed at the IoT edge that enables compatibility between industrial or other SCADA-type hardware and the device management and data management platform components.

Rules Engine (Historical)

Rules engine (historical) is the ability to execute actions including external callouts, notifications and alerts based on stored machine data. The actions performed are based on machine data that have been stored. The rules engine (historical) can either be based on anomaly-detection rules, moving averages or other operator- or administrator-defined parameters.

Rules Engine (Real-time)

Rules engine (real-time) is the ability to execute actions including external callouts, notifications and alerts based on live or streaming machine data. The rules engine (real-time) can also provide anomaly-detection and value limits, but these must be provided in near real-time with event processing occurring within a few minutes from initial data ingestion.

Software and Firmware Repository

Software and firmware repository is a service that provides a centralized collection point for software and firmware to be pushed to or accessed directly by IoT devices, aggregation points or gateways.

Southbound Data APIs

Southbound data APIs enable communication on the data layer between connected IoT devices, aggregation points and gateways and data ingestion and routing service components. Southbound data APIs are typically MQTT/HTTP(S) endpoints, but many different protocols are used in different platforms.

Southbound Management APIs

Southbound management APIs enable bidirectional management layer communication between a device management service and managed IoT devices, aggregation points and gateways. Southbound management APIs are often provided as an HTTPS endpoint, but proprietary protocols are also common. These APIs are distinct from the machine data ingestion endpoint in that no actual machine data is provided over this channel, only data associated with device management such as lifecycle management commands, firmware updates, and other device management functionality.