ActionVector IoT

THE SMART IOT PLATFORM

TERMS AND DEFINITIONS

Table of Contents

Introduction

This document describes important terms and conceptual ideas that are central to the IoT platform design. This is a work-in-progress document.

Reader is advised to read this document before the other documents.

Device

A device is any physical device that is a part of a network of devices managed by the IoT Platform. A device is anything such as an electric bulb, microwave oven, an energy meter, a solar inverter, a computer, a car, a motor, a refrigerator, or even a software component or anything that participates in an IoT network. Every device is associated with a vendor. Every vendor may provide one or more such devices.

Any device, in a generic sense can be of type sensor, actuator or both or, even a proxy type. A proxy type refers to a device that serves as hub or a gateway for one or more devices.

A user is the user of various services (capabilities) provided by a device. Device owner is a user with full access to all device capabilities. A device owner is owns the support contract and other documents related provided by the vendor. An end user is a normal user of a device. A device also provides administrative and support services that are used by a support personnel to manage the device. A virtual user is any other software service or an application that accesses a device. A security profile defines the access to various device services by different types of users.

Every device supports some functionality that is central to the device. A service is simple abstraction of such services provided by a device. For example, a refrigerator provides a refrigeration service, an inverter provides DC-AC conversion service, a Temperature sensor provides a temperature measurement service and so on.

A smart device, provides additional administrative services to control and monitor the device by an external system or human agents (user). For example, a user or a system may send a command to the device to get the state of the device. The user or system may send a series of control commands to modify its state. For example, one could visualize a Television set providing control services such as power on service, power off service, channel navigation services to select a channel, volume control services, auto scan service to detect channels, configuration services to set various operating parameters etc. and status query services that provides the status of the device in terms of a set of variables such as currently active channel, current volume level, amount of power drawn etc.

Similarly one could visualize the capabilities provided by all devices in terms of services. It is easy to imagine the capabilities provided by any other device such as microwave oven, a car, a computer, etc. in terms of user services and admin services.

The devices can be organized into hierarchical groups for administrative convenience or necessity. For example, a solar plant or an installation can be modeled as a device group comprising of all solar devices in the site, or it can also be used to manage hierarchical sub-installations to manage and control smaller groups of related set of devices. Devices within a group may also interact with each other's services. For example, IoT enabled vehicle may share its location information with nearest traffic signaling system in traffic management system. Devices may be grouped in order to view and analyze the performance of a sub-system at a desired level of aggregation in the performance reports.

The Communication Interface

The smart devices support a set provide a mechanism for bi-directional interaction between the device and external system/user.

The communication interface is the means to interact with the device to issue control commands or retrieve device state information. An Interface is in the form of a hardware (physical communication interface), and software protocol. The physical communication interface may be a wired or wireless interface. A communication protocol can be visualized as a set of messages used to communicate to/from the device as per a well-defined standards based or proprietary protocol on physical communication interface. For example, Modbus/TCP, Modbus/HTTP, Zigbee, Canbus, etc. The various control buttons in a TV set are a good examples of hardware/physical interface or the electrical and electronics circuit in a TV that responds to the commands from a remote is an example of a hardware interface. Many modern devices including electrical (e.g. inverter, energy meters, motors) and electronics devices provide standard communication interfaces such as RS 232, RS485, Ethernet, WLAN with software protocols to invoke the control and status services.

The communication protocols specify the message structures, data types, programming interfaces to enable software programs written in different languages to interact with the devices. A communication protocol may be synchronous or asynchronous.

Command

A data structure that is used to invoke an operation on a device.

State

A data structure that is used to represent the state of a device as a set of parameters.

Event

State of a device at a particular instant of time is defined as an event. The state of a device is represented by an appropriate set of parameters. For example, the state of smart inverter can be defined by voltage, current, power and energy, state of motor can be defined by speed, voltage, current, on/off status etc.

Event Stream

A continuous stream of events emitted/received by any component.

Device Schema

Software programs written in specific language interact with the smart devices through the communication protocol messages to send commands and retrieve the state information. A device schema is the metadata that is used to define commands and event data structures of a specific device. The Device Schema varies across different types of devices and device vendors.

Common Platform Schema

A common Platform schema is a generic metadata that is used to define critical data structures such as generic device commands and events common to the IoT platform and not specific to device types and vendors. The common platform schema enables IoT platform components to deal with various data elements associated with devices in a uniform and consistent way across device types and vendors.

Schema Mapper

A schema mapper is a software component that maps the device message schema to common domain schema.

The IoT platform comprising of many components need to deal with the command and event structures that are independent of any one device type or vendor. For example, name of a parameter such as voltage may be spelt differently in different device types and/or vendors. Similarly, a command to switch off a device may be represented differently in different device types by different vendors.

Schema mapper enables mapping device type/vendor specific representations of command and state data structures to uniform and consistent internal representation for processing within IoT platform. This enables easy integration of new device types and devices from new vendors into the Io T platform.

Event Processor

An Event Processor, receives an evet stream in real time and applies a set of rules to perform operations such as event filtering, aggregations, pattern detection, event correlation etc. and dispatches the outcome as an event to an end consumer (configured endpoint) for further action. Event Processor is often referred to as stream analytics engine or Complex Event Processor (CEP).

Event Logger

An event logger receives the raw event from the device and logs the events in a repository after performing operations such as cleaning, filtering, transformation operations. The event repository is used for data analytics and reports.

Device Owner

An individual or a business entity who owns the device.

Device user

An individual or a business entity who uses the device

IoT Infrastructure Provider

An individual or a business entity who designs, deploys and operates the IoT infrastructure including devices, communication networks, servers etc.

IoT Service Provider

An IoT Service Provider is a business entity who delivers end consumer applications and services on the network of devices (IoT).

IoT Service

An application or a service Provided by the IoT Service provider to the IoT Consumers. A Service provider provides one or more services to the consumers on IoT infrastructure.

IoT Consumer

A IoT consumer is a business entity or an individual who is the end consumer or beneficiary of the applications and services provided by the service provider.

Customer

A customer is a business entity who owns the IoT. For example, a business entity which invests in a solar plant comprising of solar panels, inverters, and other devices and generates and delivers electrical energy to utility companies is a customer or a business entity that owns an IT Infrastructure such as an enterprise or a data center is a customer.

Data Aggregation Dimensions

The data is primarily aggregated along two dimensions namely device hierarchy and time hierarchy. The device hierarchy comprises of three level hierarchy of device, device group and customer and time hierarchy comprises of time dimensions namely – hour, day, week, month, year or any user defined period. The various metrics from devices can be aggregated and analyzed (slicing and dicing) along device and time dimensions. This model represents what is currently supported in the IoT Platform. However, any third party analytics platforms and tools can also be easily adapted.

Basic Metric type

A basic metric type is the metric type that is natively supported in the device and made available through the state event.

Derived Metric type

A derived metric type is a secondary metric that is derived from one or more basic metrics types. A derived metric generator object encapsulates the necessary logic to derive a new metric from the available data in the event repository. Generating statistical metrics such as statistical average, min, max, standard deviation of a data set are some examples of derived metrics. A derived metrics generator may also have any proprietary logic to derive new metrics from the available data in event repository.

Analysis View

An analysis view is a model used to describe the behavior of the devices or device groups along various dimensions. The Analysis View comprises of two parts namely dataset and graphical visualization (chart) of the dataset.

Data Set

The data set is a set of data values of a certain metric type (basic or derived) aggregated over a specified level in the device and time hierarchy over a defined period of time. The datasets are designed to support some of the commonly used analysis views such as time series, comparisons, deviations, correlations, ranking, whole part, frequency distribution etc. and, are amenable for visualization via suitable types of charts such as line charts, bar chart, time charts, grouped bar charts, pie charts etc.

Chart Generator

A chart generator generates the data set based on the analysis model defined and, an external GUI application uses the data set to provide a corresponding graphical display using appropriate chart controls.

IoT Application

An IoT Application is a domain specific, horizontal or vertical applications. The Application services bring together the aggregated capabilities of a network of devices and enable remote management of the network of devices. The application may be cross-domain or local to a single domain. These applications encapsulate business logic or business rules to interpret the data from the devices and initiate appropriate actions.

Installation

An installation is a named group of devices in a site or a location. It is used to identify the site or the location where one or more devices are deployed and running.