OCF oneIoTa Frequently Asked Questions

What is the Theory of Operation for the OCF architecture?

The Open Connectivity Foundation is an organization devoted to the interoperability of devices in the Internet of Things. It is a RESTful architecture based on CRUDN that securely communicates using simple, open data structures that describe basic resources and the devices that are composed of these resources. It achieves interoperability using simple open definitions that describe the mapping between devices in different ecosystems. It can map to other RESTful systems as well as systems based on more procedural architectures and remote procedure calls (RPC). oneIoTa is an open tool for developing and managing the open resources of OCF and other organizations as well as the mappings between them. OCF includes a rigorous certification program and an automated test tool. OCF also supports the IoTivity open-source project that is a certified implementation of the OCF architecture.

What is a RESTful architecture?

RESTful stands for Representational State Transfer and is an design method for stateless network communications. (REST) There are instructional materials and tutorials available from many sources on the web. OCF uses CRUDN, which adds notification to the basic REST architectural model.

What is CRUDN?

CRUDN stand for Create, Read, Update, Delete and Notify. These are the five operations in the RESTful architecture used in OCF.

What makes OCF secure?

OCF describes a very complete security environment based on hardware-level credentials and certification. Devices from other ecosystems that don't meet the OCF security requirements will be limited to certain safe interoperability scenarios.

What is IoTivity?

IoTivity is an open-source project hosted by the Linux Foundation and supported by OCF. It is a certified OCF implementation that can be run on an increasing number of hardware and software platforms.

What is a resource?

A resource is a minimal interoperable component in OCF that is intended to be reused in multiple contexts. For example, a binary switch is a resource. In OCF, resources are defined using JSON. All OCF resources are publicly available in oneIoTa (at oneIoTa.org) and from the associated git repository. A specific collection of resources can be composed into a device. If a specific collection of resources is likely to be frequently reused (For example, a thermostat with a set temperature, an ambient temperature, and a switch), it can be composed into a complex

resource. Resources can be mapped to models from non-OCF ecosystems using Derived Models. Thus OCF resources are the fundamental building block of OCF interoperability.

What is a device?

A device is a specific collection of resources composed to meet the needs of real-world devices. Any developer or manufacturer can define a device using OCF resources. It will be interoperable with other devices that have resources in common. Interoperability doesn't require a one-to-one correspondence between devices. For example, refrigerators from two manufacturers may have several different, proprietary features. However, if they have resources for a vegetable crisper and a freezer in common, those resources can be controlled by a common controller or data collected from the two refrigerators can be directly compared.

What is OCF bridging?

OCF bridging is a way to make different ecosystems interoperable by connecting them through OCF. A bridge will implement the mappings between OCF and any other ecosystem using OCF derived models. A single mapping from an ecosystem to OCF will enable interoperability with all other ecosystems that map to OCF. The bridge will also handle any communication mappings at the lower levels of the networking stack. The lower level mapping are not within the scope of OCF.

What are the models?

In oneIoTa, there are three types of models.

- Native Models
  Native Models are OCF defined generic resources that can be reused to describe any device. They include things like switches, thermometers and sensors. Atomic resources can be combined in collections of resources to make more complex resources. However, it is important to make a distinction between resources and devices. Devices are models of something a manufacturer may create and sell. Resources are the individual components that make up a device and are meant to be reused regardless of the context. So, for instance, a thermostat resource might be used in a home heating thermostat, a car radiator or maybe two of them in a refrigerator/freezer device. Since OCF is designed to be interoperable on a resource level, only the resources need to be mapped to other ecosystems to ensure common communication interoperability of arbitrarily complex devices. Devices provide the structure and context definition for interoperability.
- External Models
  External Models are resources contributed into oneIoTa by SDOs and other organizations for interoperability and the good of all. These models are approved and maintained by the External Organization. These models are not directly used by OCF, but Derived Models describe the mapping between them and Native Models. If there is not an equivalent Native Model resource to match and External Model, the External Organization should create one and submit it for approval to OCF along with a Derived Model to describe the mapping between them.

- Derived Models
  Derived Models describe the mapping of resources between External Models and Native Models. Derived Models are approved and maintained by the External Organization.

How does oneIoTa fit into the OCF Theory of Operation?

oneIoTa is a shared online crowdsourcing tool for defining data models that describe the basic resources for the Internet of Things. It solves the interoperability issue for the Internet of Things without requiring organizations to change their existing approach. oneIoTa combines the basic features of an integrated development environment for data models with an automated approval process that is managed by the organization sharing the models. The models stored in oneIoTa are then used to automatically generate documentation, code stubs and test scripts for certification. This simplifies the development process, speeds up prototyping and makes everything more robust.

What is an OCF Native Model?

Native Models are OCF defined generic resources that can be reused to describe any device. They include things like switches, thermometers and sensors. Atomic resources can be combined in collections of resources to make more complex resources. However, it is important to make a distinction between resources and devices. Devices are models of something a manufacturer may create and sell. Resources are the individual components that make up a device and are meant to be reused regardless of the context. So, for instance, a thermostat resource might be used in a home heating thermostat, a car radiator or maybe two of them in a refrigerator/freezer device. Since OCF is designed to be interoperable on a resource level, only the resources need to be mapped to other ecosystems to ensure common communication interoperability of arbitrarily complex devices. Devices provide the structure and context definition for interoperability.

What is a derived model?

Derived Models describe the mapping of resources between External Models and Native Models. Derived Models are approved and maintained by the External Organization.

Who owns the models in oneIoTa?

Native Models are owned by OCF. External Models are subject to the terms of the agreement made between OCF and the sourcing organization. Derived Models are owned by OCF. All models publicly exposed in oneIoTa are available for anyone to use subject to the oneIoTa terms of use.

oneIoTa doesn't seem to have the native resource I need. What should I do?

If the resource model you need isn't defined as a Native Model and it is a model that is likely to be reused by others to create devices, you can create it as a proposal in oneIoTa (either by combining existing OCF resource models or adding a brand new one) and submit it to OCF (using the submit button in oneIoTa). The proposal will be evaluated by the OCF Data Model Working Group and either be accepted as a Native Model, returned to you with comments

about why it wasn't accepted, or rejected (because it doesn't meet the requirements for a Native Model, or duplicates a model already approved).

How do I get my favorite SDO registered and involved in oneIoTa?

SDOs interested in becoming an approved organization in oneIoTa for contributions and approvals of that SDO's data models should contact staff@openconnectivity.org for more information. One of the first steps will be to provide the SDO's proposed licensing terms for data model contributions. Upon approval, it will be necessary to provide the SDO's contact persons to receive notifications of new data models submissions, and the github link for where the data models will reside.

How do I submit a derived model for my favorite SDO?

Follow the same process described above for the submission of a Native Model, but submit the proposal to the specific SDO instead of OCF using the submit button in oneIoTa. NOTE: the SDO must already have an agreement with OCF and be integrated into oneIoTa.

Do I have to use Apache-2 as dictated by membership in OCF or can I choose a different license?

Each organization that decides to work with OCF will create a bilateral agreement with OCF that will specify the relevant contributor agreements. In oneIoTa, a user will be asked to agree to the relevant terms per organization.

Do I own my entries into oneIoTa or does OCF?

Each organization will continue to own and maintain its data models. However, upon making an agreement with OCF the terms of submitting models will be specified.

Can OCF use the information I enter into oneIoTa without my permission?

Models entered in oneIoTa are publicly available to anyone. Specific terms between an organization and OCF will be agreed before working with OCF.

Can OCF change what I put into oneIoTa?

Anyone can make a proposal, but the proposal will not be public until it is accepted by the organization to which it was submitted. OCF is the approving organization for OCF native models. Other models will be approved by the appropriate organization (i.e. Not OCF).