## A WoT gateway with device virtualization

Ryuichi Matsukura and Takahisa Suzuki, Fujitsu Laboratories Ltd.

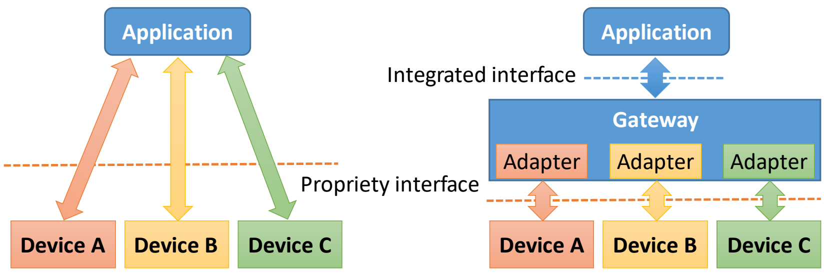
Takuki Kaimiya, Fujitsu Laboratories of Amerika Inc.

*r.matsukura@fujitsu.com*

### 1. Introduction

Many devices in the various fields have been connected to the network and are required to be accessible and controllable from the cloud to establish the digital transformation. However, these devices are usually connected not in a limited standard but in combination of some standards. The Web of Things (WoT) is an idea to integrate the connectivity of the devices that support diverse network media, transport protocols and information models to a standard way using with Web based interface.

Figure 1 shows how the applications connect the devices in multiple interface standards. In Figure 1(a), the application connects three different devices with different interfaces. Although it's the simplest way, the application needs to support different interfaces in the same time and requires a lot of knowledge to application developers. Especially, connecting the devices using old standards that do not support IP is not familiar for many developers. Therefore, as shown in Figure 1(b), it could be better to convert the interface for each device to an integrated interface on a gateway. WoT is suitable as this interface. WoT can also provide a developer-friendly interface to connect application and devices to each other through the Web based interface.




(a) Multiple interfaces (b) Integrated interface on Gateway

Figure 1. IoT system diagrams focusing on device interfaces

This paper describes a gateway that can virtualize capabilities of various IoT devices based on the specification of W3C WoT. The gateway can aggregate other gateways upstream to increase the number of connected devices and broaden areas. In the past WoT plugfests, our gateway based on this idea can connect many devices provided by WG members provided and can be controlled from some applications they did. In addition, we have started to evaluate our implementations including gateways and devices in some fields such as smart home, smart factory and smart agriculture, in which over 1,000 devices have been connected.

### 2. Gateway

In WoT, the devices are virtualized and their capabilities are represented as JSON objects based on the Thing Description (TD). The JSON objects are handlesd with REST. Figure 2 illustrates an example of a representation for Air Conditioner on the gateway and how the application handles the device in the WoT manner. The table inside the virtual device is an information model corresponding to the physical device capabilities described with JSON based on TD. The Properties are virtualized capabilities the physical device has. For example, "Status", "Mode", and "Temperature" mean the power status (on / off), the operation mode (cooking / heating) and the set temperature respectively. The virtual device can be handled with some methods of a transport protocol such as READ, WRITE and NOTFY. READ or WRITE are methods to get or change the current value of that property specified with. NOTIFY is a method to automatically notice the current value from the device when the value is changed. These methods are assigned to HTTP methods in RESTful manner so that the application changes the JSON object and then the physical devices can be operated by the gateway through the propriety interfaces of them.

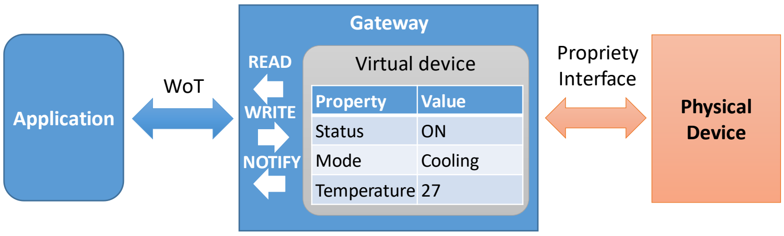



Figure 2. Virtualized device on gateway

For example, BACnet, KNX and ECHONET Lite that are standards for devices of buildings and houses have a similar structure and contents for information models and operation methods to WoT. However, since their representations and communication media are different from WoT, some adapters that can convert their models and method to WoT are required to the gateway.

The devices connecting ways to the gateway are classified into three patterns depending on the interfaces of the devices as shown in Figure 3. If the device is compatible to WoT (WoT device), it can directly connect the gateway through the IP networks (Figure 3(a)). Otherwise (non-WoT device), the adapters are needed between the gateway and the devices. If the device interfaces are widely used and supported by the gateway that is PC or Linux mini PC, the gateway should connect the device through the propriety interface and convert its interface with the adapter on the gateway (Figure 3(c)). If the interface is not used widely, the external adapter is better to be introduced near the devices to connect the gateway and device through WoT (Figure 3(b)).

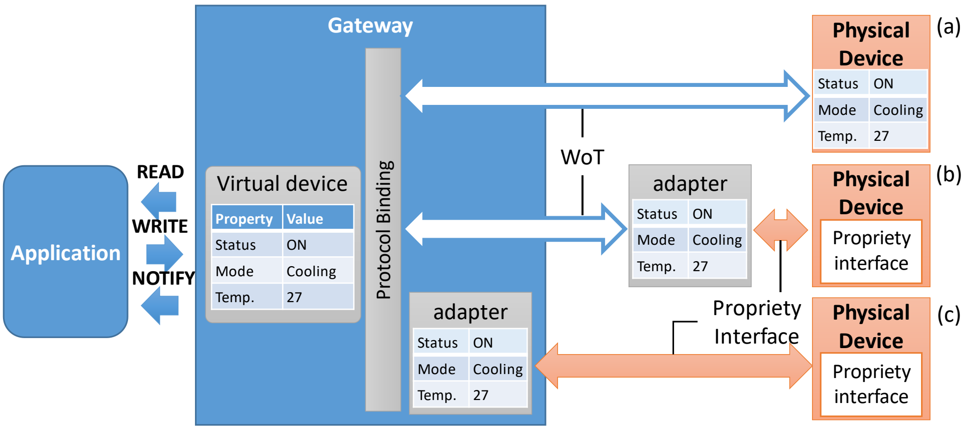



Figure 3. Three patterns to connect gateway and devices

## 3. Implementation

This chapter describes our implementation of the gateway supporting WoT. As mentioned above, non-WoT devices could be converted to WoT representations through the adapters so that any devices could be represented as the virtual WoT devices on the gateway.

When the gateway can detect a newly connected device through the interface, it starts a process to create the virtual device. If the device to be connected is the WoT device, the gateway gets TD from the device or some directory services described later. If not, the gateway installs the adapter corresponding to the interface and the adapter generates TD based on the information model got from the device. The gateway can expose the TD of the virtual device to others instead of the physical device after TD generations. Since these operations are almost automatically proceeded on the gateway regardless of whether the device supports WoT or not, it means the device are available with zero configuration right after the connection.

The gateway has been implemented in JAVA and manages modules using OSGi framework. Since the adapters are developed as bundles (plug-in), they can be installed easily from the management function running on the outside. We have developed 20 types of adapters corresponding to the standard interfaces of the devices for the field experiments described later. Because these bundles are almost applicable to the devices support the same standards, it's easy to reuse for the other devices.

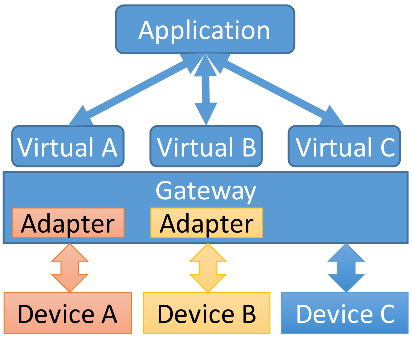



Figure 4. Virtual device lifecycle management

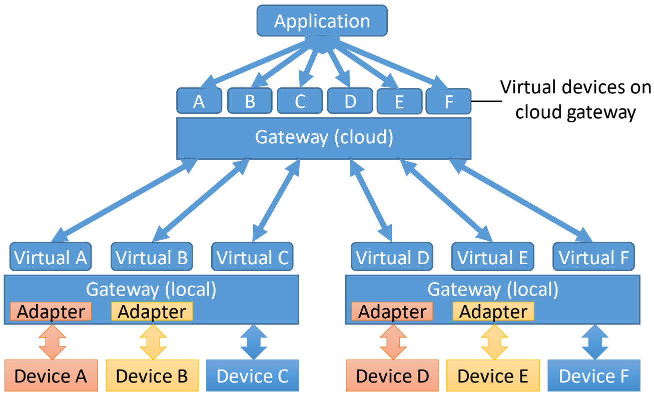



Figure 5. Multi-layered gateways for practical usages

This gateway can aggregate other gateways upstream as shown in Figure 5 so that the number of the connected device could increase and the areas could broaden by adding gateways downstream. For example, for smart homes, the gateway on the cloud aggregates the gateways that connect the home appliances and sensors deployed in houses. In addition, a smart grid service that enhances balances of power consumptions and generations runs on the cloud gateway and controls the devices deployed in each home and building. An another examples is for smart factories that multiple local gateways that cover a large space of the factory can connect huge number of devices.

The lower layer gateways that connect the physical devices are aggregated with the upper layer gateways so that all of the devices connected to the lower gateways could be controlled by the upper gateway. Since the virtual devices generated on the lower gateways are regenerated as other virtual devices on the upper gateway, the application on the upper layer gateway can handle the physical devices through the upper and lower gateways. This architecture might be good for the developers to make managing devices easier, but not good for users to make performance lower. If high performance is more important, this architecture can deploy some parts of applications that require quick responses on the lower gateway. This means to realize edge computing in this architecture. Both virtual devices on the lower and the uppers correspond to the same physical devices, but they are different entities that have different addresses or ID. However, if two or more gateways control the same physical device, it is necessary to mediate control between gateways.

Since the upper layer gateways don't need to connect devices directory in many cases, it can run on the servers or clouds that don't have hardware interfaces to connect the devices. On the other hands, the lower layer gateways require hardware interfaces like not only Ethernet but also radio interfaces such as Bluetooth, serial interfaces such as RS-485, or interfaces for industrial facilities.

## 4. Evaluations in the fields

We have already started to evaluate our system including gateways and many kinds of devices in the fields as shown in Figure 6, e.g. a home, a factory and agriculture fields like greenhouse horticulture. These systems are composed of two layered gateways that connect the applications on the cloud and the devices deployed on the local networks. For the experiment in the home, over 200 devices, e.g. home appliances, power equipment like solar panel and storage batteries, home facilities like blinds and windows and sensors, are connected using ECHONET Lite, which is an international standard for home appliances and much used in Japan. All of the devices are accessible and controllable from the cloud as shown in Figure 6(a).

For the experiment in the factory, about 40 sensor units are connected to 4 gateways in the field of 6,000 sq. meters. Each sensor units have 6 sensors to monitor the environmental conditions and a Wi-Fi communication module equipped with the WoT stack. This WoT stack has some extensions that can negotiate with the gateway to register own TD and initiate the virtual device on the gateway. This extension can realize zero configuration.

For the experiment in the agriculture fields, 150 sensor units are installed to 5 fields throughout Japan to keep a good environment for plant efficient growth in the horticulture. The sensors are designed to operate for three years without externa power supplies, as there is limited to access the power supplies in the fields. These sensors equip solar panels and storage capacitors and use Bluetooth Low Energy (BLE) for low power radio communication to provide 24-hour monitoring.

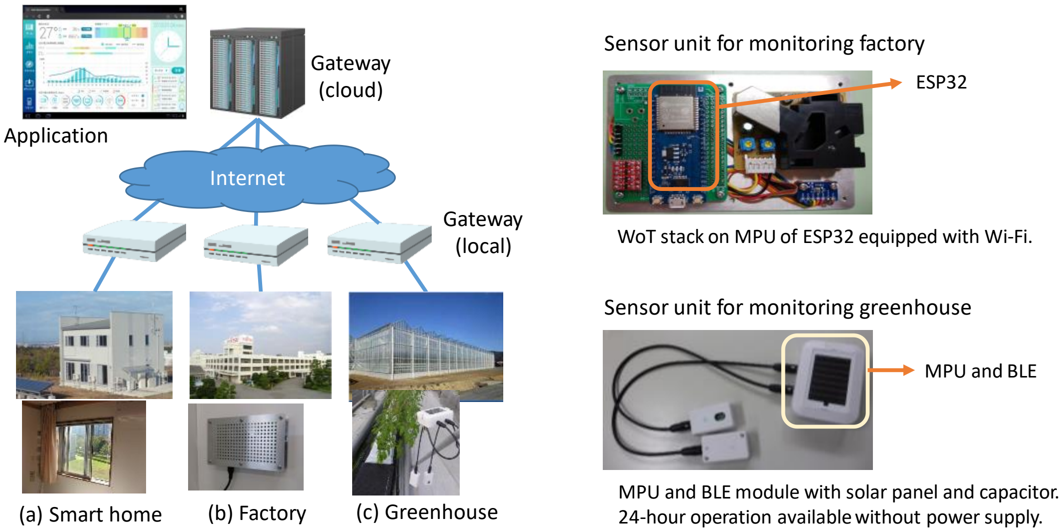



Figure 6. Current experiment fields and Sensor units

## 5. Conclusion

This paper described our implementations and their system evaluation in three kinds of experimental fields. Some of them have continued for more than a year. Over 1,000 devices have been connected throughout Japan and are aggregated in the cloud.

The gateway supports WoT specification and extends some features for enhancement;

- Has adaptation framework to expose the existing devices as a WoT device
- Can expose the integrated interface compatible to WoT regardless of whether the device supports WoT or not
- Can aggregate the lower layered gateways to connect huge number of the devices and cover a large area.
- Can manage the connected devices as virtual devices and provide the interface to search them.


## Acknowledgement

The result of the research and development was conducted as a part of the project funded by the Japanese Ministry of Internal Affairs and Communication.