OCF 2.0 – BLE Translation – Bridging Task Group CR 2509

Legal Disclaimer

1 Scope

This document specifies a framework for translation between OCF devices and other ecosystems, and specifies the behaviour of a translator that exposes servers in non-OCF ecosystem to OCF clients and/or exposes OCF servers to clients in non-OCF ecosystem. Translation per specific device is left to other specification (deep translation). This document provides generic requirements that apply unless overridden by a more specific document.

2 Normative references

Adopted Bluetooth Profiles, Services, Protocols and Transports
https://www.bluetooth.com/specifications/adopted-specifications

Bluetooth Core Specification 4.0 or higher
https://www.bluetooth.com/specifications/bluetooth-core-specification

OCF Core Specification, Open Connectivity Foundation Core Specification, Version 1.3
https://openconnectivity.org/specs/OCF_Core_Specification_v1.3.0.pdf

OCF Resource Type Specification, *Open Connectivity Foundation Security Specification*, Version 1.3
https://openconnectivity.org/specs/OCF_Resource_Type_Specification_v1.3.0.pdf

OCF Bridging Specification, *Open Connectivity Foundation Bridging Specification*, Version 1.3
https://openconnectivity.org/specs/OCF_Bridging_Specification_v1.3.0.pdf

OCF Security Specification, *Open Connectivity Foundation Security Specification*, Version 1.3
https://openconnectivity.org/specs/OCF_Security_Specification_v1.3.0.pdf

IETF RFC 4122, *A Universally Unique IDentifier (UUID) URN Namespace*, July 2005
https://www.rfc-editor.org/info/rfc4122

IETF RFC 7231, *Hypertext Transfer Protocol (HTTP/1.1): Semantics and Content*, June 2014
https://www.rfc-editor.org/info/rfc7231

3 Terms, definitions, symbols and abbreviations

3.1 Terms and definitions

3.1.1 Symmetric, Asymmetric Bridging

In symmetric bridging, a bridge device not only exposes OCF server to other ecosystem but also exposes other ecosystem's server to OCF, on the other hand, in asymmetric bridging a bridge device exposes OCF server to other ecosystem only or exposes other ecosystem's server to OCF only.

3.1.2 Bridged Protocol

Another protocol (e.g., AllJoyn, BLE) that is being translated to or from OCF protocols

3.1.3 GATT-based Profile

A BLE profile using procedures and operating models provided by GATT profile

3.2 Symbols and abbreviations

3.2.1 GATT

Generic ATTribute profile
Generic Attribute Profile describes a service framework using the Attribute Protocol for discovering services, and for reading and writing characteristic values on a peer device (vol 3 part G of Bluetooth Core Specification).

3.2.2 ATT

ATTribute protocol
ATT is a protocol for discovering, reading, and writing attributes on a peer device (Vol 3, part F of Bluetooth Core Specification)

3.2.3 GAP

Generic Access Profile

GAP defines the generic procedures related to discovery of Bluetooth devices (idle mode procedures) and link management aspects of connecting to Bluetooth devices (connecting mode procedures). It also defines procedures related to use of different security levels. In addition, this profile includes common format requirements for parameters accessible on the user interface level. (vol 3, part C of Bluetooth Core Specification)

8 BLE Translation

8.1 Operational Scenarios

The overall goal is to make Bridged BLE GATT Servers appear to OCF Clients as if they were native OCF Servers in the local network or cloud environment.

"Deep translation" between specific BLE Profile and OCF Device is specified in a separate document. Figure 1 shows overview of BLE Bridge device and its general topology. The BLE Translator supports Asymmetric bridging. It exposes BLE GATT Servers to OCF Clients. Each Bridged BLE GATT Server is represented as a Virtual OCF Server.

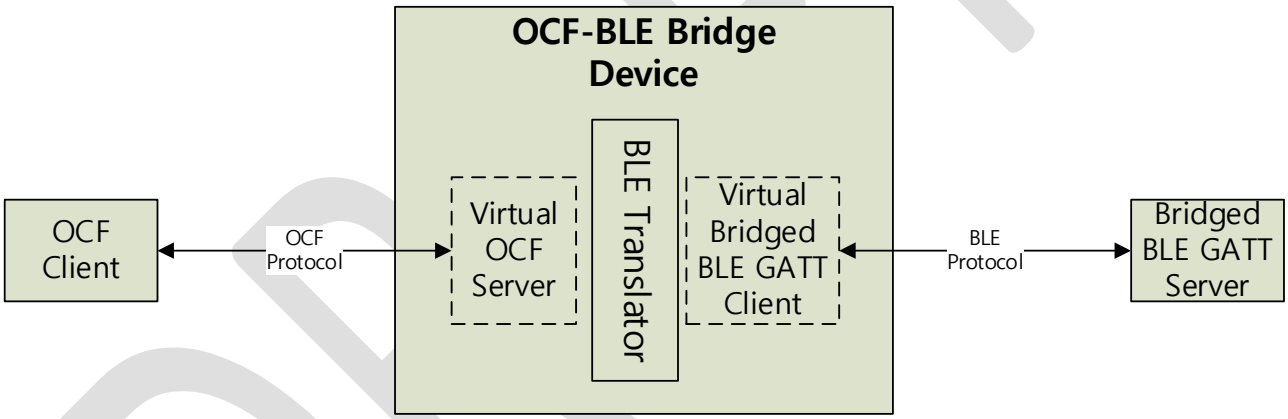


Figure 1 OCF-BLE Bridge Device Components

8.1.1 Use case for BLE Bridging

Figure 2 shows a use case for an OCF Client and BLE GATT Server. An OCF client on a smartphone reads a BLE thermometer device through an OCF-BLE Bridge. Any connectivity that OCF supports is used for communications between the OCF Client and the OCF-BLE Bridge. The OCF Client can communicate with OCF-BLE Bridge through OCF Cloud.

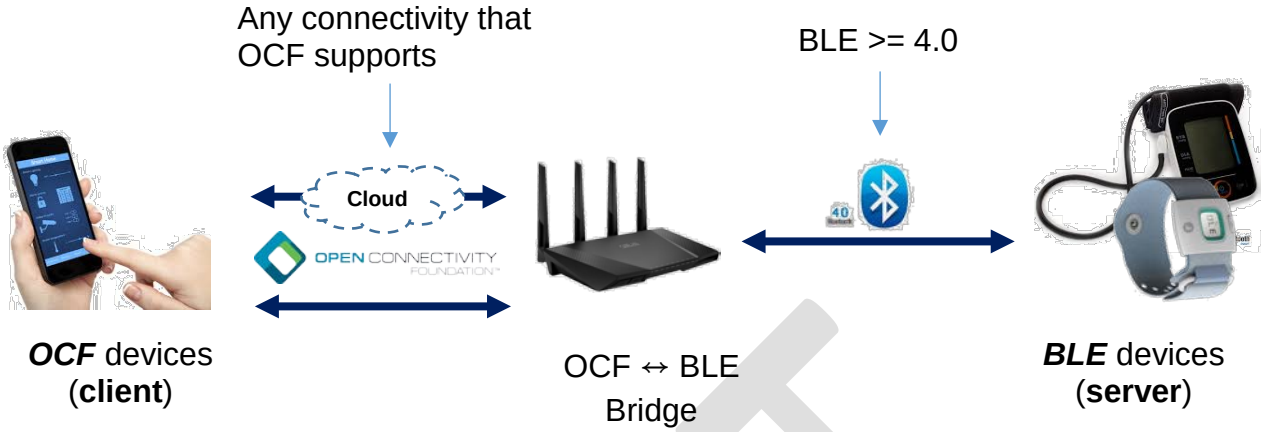


Figure 2 BLE Bridging use case in real life

8.2 Requirements specific to BLE Translator

OCF-BLE Bridge device shall satisfy section 5.2 General Requirements of OCF Bridging Specification.

A BLE translator supports asymmetric bridging. It exposes BLE GATT server to OCF clients only. Therefore, it shall play a BLE GATT client role. (This is a requirement so that users can expect that a certified OCF Bridge Device will be able to talk to any BLE GATT server device, without the user having to buy some other device.).

8.2.1 Requirements specific to BLE

The version of Bluetooth SIG core specification that this document refers to is 4.0 or higher. Bluetooth BR/EDR is not included in the scope of this document.

8.2.2 Exposing BLE GATT Servers to OCF Clients

The requirements in this section apply when using algorithmic translation, and by default apply to deep translation unless the relevant specification for such deep translation specifies otherwise.

Basic translation rule between BLE Service/Characteristic model and OCF Resource model is described in Table 1. One or more BLE GATT-based profiles should be mapped to one Virtual OCF Server (e.g. Health Thermometer profile (HTP) is mapped to Body Thermometer device (oic.d.bodythermometer)). A BLE Service should be mapped to one or more OCF Resources (e.g. Health Thermometer Service is mapped to Temperature (oic.r.body_temperature) and Body Location for temperature (oic.r.body.location.temperature)). Each Characteristic of BLE Service should be mapped to one or more Properties of OCF Resource (if there is no BLE Characteristic corresponding to an OCF Property, default value should be used). Table 2 is a translation example of this rule.

Table 1 Translation rule between BLE and OCF data model

From BLE	mapping count	To OCF	mapping count
GATT-based profile	**n**	OCF Device	**1**
Service	**1**	OCF Resource	**n**
Characteristic	**1**	OCF Resource property	**n**

Characteristic Descriptor	**1**	OCF Notification on/off option	**1**

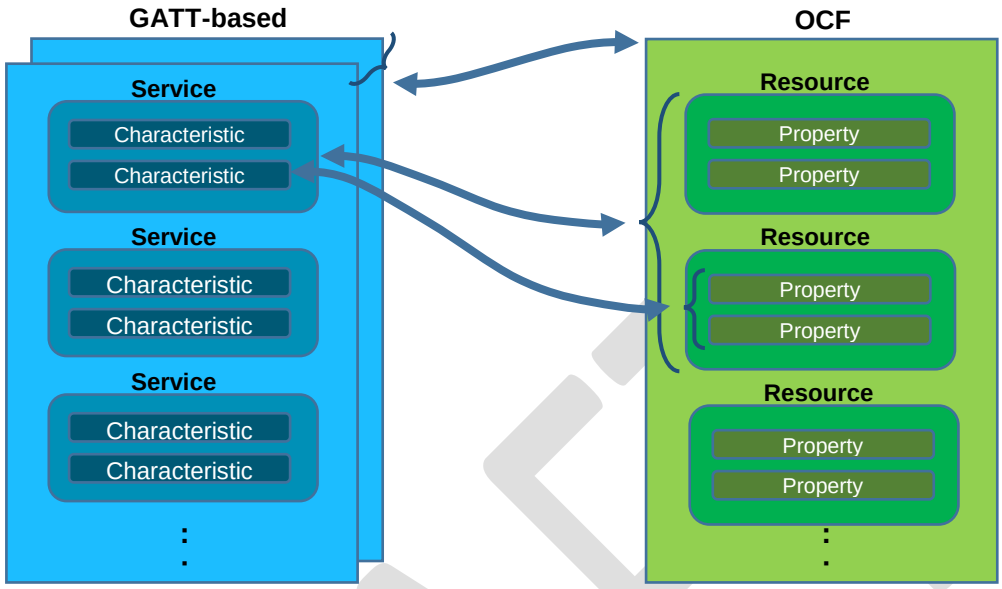
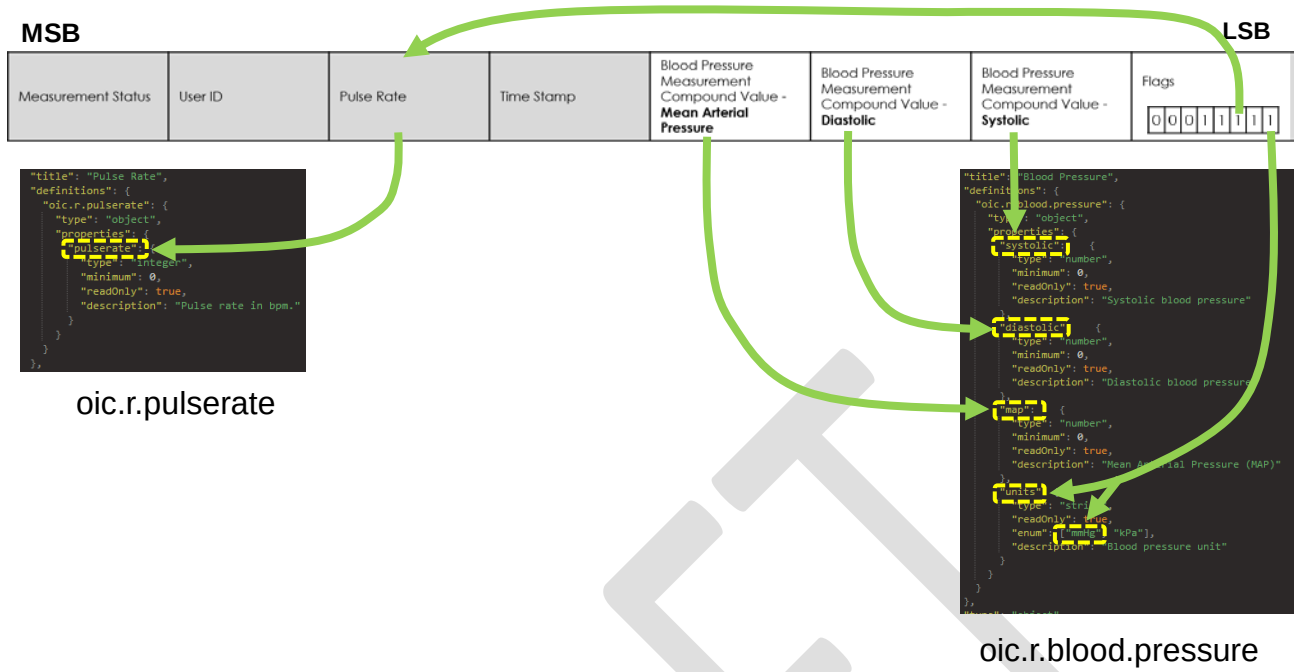


Table 2 BLE → OCF translation example (Blood Pressure device)

	BLE	OCF
BLE Profile → OCF Device	Blood Pressure Profile (BLP)	Blood Pressure Monitor device (oic.d.bloodpressuremonitor)
BLE Service → OCF Resource	Blood Pressure Measurement Service (org.bluetooth.service.blood_pressure)	Blood Pressure (oic.r.blood.pressure)
		Pulse Rate (oic.r.pulserate)
	Device Information Service (org.bluetooth.service.device_information)	Device (oic.wk.d) Platform (oic.wk.p)
BLE Characteristic → OCF Resource Property	Blood Pressure Measurement (org.bluetooth.characteristic.blood_pressure_measurement)	oic.r.blood.pressure.systolic oic.r.blood.pressure.diastolic oic.r.blood.pressure.map oic.r.blood.pressure.units
		oic.r.pulserate.pulserate

Figure 3 shows an example for 1:N mapping between “BLE Characteristic” and “OCF Properties”. In this case, multiple fields in “Blood Pressure Measurement Service” are mapped into the properties of OCF Resources (oic.r.pulserate, oic.r.blood.pressure).



157

158 **Figure 3 An example for 1:N mapping between "BLE Characteristic" and "OCF Properties"**

159 **8.2.2.1 Translation for well-defined set**

160 If a BLE Profile is in a well-defined set (defined in Device Mapping specification ("deep translation")),
161 translation should be done as follows. Table 3 is the list of BLE GATT-based Profiles which have
162 corresponding OCF Resources as of now.

163 **Table 3 BLE GATT-based Profile – OCF Resource mapping**

BLE GATT-based Profile	BLE Service	OCF Resource		OCF Device Type
		Atomic Measurement Resource Type	Resource Type	
Blood Pressure Profile	Blood Pressure Service	oic.r. bloodpressuremonitor-am	oic.r.blood.pressure	oic.d.bloodpressuremonitor
			oic.r.pulserate	
	Device Information Service		oic.wk.d	
			oic.wk.p	
Glucose Profile	Glucose Service	oic.r.glucosemeter-am	oic.r.glucose	oic.d.glucosemeter
			oic.r.glucose.carb	
			oic.r.glucose.exercise	
			oic.r.glucose.hba1c	
			oic.r.glucose.health	
			oic.r.glucose.meal	
			oic.r.glucose.medication	
			oic.r.glucose.samplelocation	
			oic.r.glucose.tester	

	Device Information Service	oic.wk.d		
		oic.wk.p		
Health Thermometer Profile	Health Thermometer Service	oic.r.bodythermometer-am	oic.r.temperature	oic.d.bodythermometer
			oic.r.body.location.temperature	
	Device Information Service	oic.wk.d		
		oic.wk.p		
Weight Scale Profile	Weight Scale Service	oic.r.bodyscale-am	oic.r.weight	oic.d.bodyscale
			oic.r.bmi	
			oic.r.height	
			oic.r.body.fat	
			oic.r.body.water	
			oic.r.body.slm	
			oic.r.body.ffm	
	Device Information Service		oic.wk.d	
			oic.wk.p	

URI for Virtual OCF Resource

- Case 1. A BLE Service is mapped to an OCF Resource:

  /<BLE Service name without prefix "org.bluetooth.service">, (e.g. BLE Service "Fitness Machine (org.bluetooth.service.fitness_machine)": /fitness_machine)

- Case 2. A BLE Service is mapped to multiple OCF Resources:

  If corresponding multiple OCF Resources are grouped by Collection (or Atomic Measurement Collection), URI should be as follows:

  a. URI for Collection Resource: */<BLE Service name without prefix "org.bluetooth.service">* (e.g. BLE Service "Health Thermometer (org.bluetooth.service.health_thermometer)": /health_thermometer)

  b. URI for each OCF Resource link: */<OCF Resource Type value of corresponding linked Resource without prefix "oic.r">* (e.g. /temperature for oic.r.temperature, /body.location.temperature for oic.r.body.location.temperature)

  If corresponding multiple OCF Resources are not grouped by Collection, URI should be as follows:

  a. URI for each OCF Resource: */<BLE Service name without prefix "org.bluetooth.service">/<OCF Resource Type value of corresponding Resource without prefix "oic.r">*

Table 4 URI mapping example

	BLE	OCF
URI	Health Thermometer Service	/health_thermometer (for Atomic Collection Resource)

	(org.bluetooth.service.health_thermometer)	/temperature (for oic.r.temperature) /body.location.temperature (for oic.r.body.location.temperature)

Common Properties of Resource Type

- Resource Type (“rt”, Mandatory): value of “rt” in corresponding OCF Resource specified in OCF Resource Type Specification.
- Interface (“if”, Mandatory): value of “if” in corresponding OCF Resource specified in OCF Resource Type Specification.

Platform Resource (“rt” == oic.wk.p)

- Platform ID (“pi”, Mandatory): since BLE device does not provide a mandatory unique “name” (or id) which can be used to generate name-based UUID described in IETF RFC 4122 section 4.3, randomly-generated UUID described in IETF RFC 4122 section 4.4 should be used for Platform ID.
- Manufacturer Name (“mnmn”, Mandatory): if Device Information Service is implemented “manufacturer_name_string” Characteristic should be used, or “<device_name> by unknown” should be used as default value (<device_name> is a Characteristic of GAP).

Device Resource (“rt” == oic.wk.d)

- Spec Version (“icv”, Mandatory): Spec version of the core specification that the translator implements should be used.
- Device ID (“di”, Mandatory): as specified in the OCF Security Specification, the value of the “di” property of OCF Devices (including Virtual OCF Devices) shall be established as part of Onboarding of that Virtual OCF Device.
- Data Model Version (“dmv”, Mandatory): spec version of the vertical specification this device data model is implemented to should be used. Syntax is “<vertical>.major.minor”.
- Protocol Independent ID (“piid”, Mandatory): randomly-generated UUID described in IETF RFC 4122 section 4.4 should be used for piid.

Exposing a BLE GATT Server as a Virtual OCF Server

Table 5 shows how OCF Device properties as specified in Table 20 in OCF Core Specification, should be derived, typically from fields specified in BLE Device Information Service (Spec Type: org.bluetooth.service.device_information, Service ID: 0x180A) and Generic Access Service (Spec Type: org.bluetooth.service.generic_access, Service ID: 0x1800).

Table 5: oic.wk.d resource type definition

To OCF Property title	OCF Property name	OCF Description	OCF Mandatory?	From BLE Device Service Characteristic value	BLE Description	BLE Mandatory?
(Device) Name	n	Human friendly name For example, “Bob’s Thermostat”	Y	Device Name (Generic Access)		Y

Spec Version	icv	Spec version of the core specification this device is implemented to, The syntax is "core.major.minor"]	Y	(none)	Translator should return its own value	
Device ID	di	Unique identifier for Device. This value shall be as defined in [OCF Security Specification] for DeviceID.	Y	(none)	Use as defined in the OCF Security Specification	
Protocol-Independent ID	piid	Unique identifier for OCF Device (UUID) . Randomly-generated UUID described in IETF RFC 4122 section 4.4 should be used for piid	Y	(none)	(none)	
Data Model Version	dmv	Spec version(s) of the vertical specifications this device data model is implemented to. The syntax is a comma separated list of "<vertical>.major.minor"]. <vertical> is the name of the vertical (i.e. sh for Smart Home)	Y	(none)	(none)	
Localized Descriptions	ld	Detailed description of the Device, in one or more languages. This property is an array of objects where each object has a 'language' field (containing an RFC 5646 language tag) and a 'value' field containing the device description in the indicated language.	N	(none)	(none)	
Software Version	sv	Version of the device software.	N	Software Revision String (Device Information)	This characteristic represents the software revision for the software within the device.	N
Manufacturer Name	dmn	Name of manufacturer of the Device, in one or more languages. This property is an array of objects where each object has a 'language' field (containing an RFC 5646 language tag) and a 'value' field containing the manufacturer name in the indicated language.	N	Manufacturer Name String (Device Information)	This characteristic represents the name of the manufacturer of the device.	N
Model Number	dmno	Model number as designated by manufacturer.	N	Model Number String (Device Information)	This characteristic represents the model number that is assigned by the device vendor.	N

Regarding configuration resource (oic.wk.con), it is not created on the Virtual OCF Server since that information/interaction is not supported on BLE side.

Table 6 shows how platform properties, as specified in Table 21 in OCF Core Specification, are derived, typically from fields specified in BLE Device Information Service and Generic Access Service.

Table 6: oic.wk.p Resource Type definition

To OCF Property title	OCF Property name	OCF Description	OCF Mandatory?	From BLE Device Service Characteristic value	BLE Description	BLE Mandatory?
Platform ID	pi	Unique identifier for the physical platform (UIUID); this shall be a UUID in accordance with IETF RFC 4122. It is recommended that the UUID be created using the random generation scheme (version 4 UUID) specific in the RFC.	Y	(none)	(none)	
Manufacturer Name	mnmn	Name of manufacturer (not to exceed 16 characters)	Y	Manufacturer Name String (Device Information). if Device Information Service is not implemented “<device_name> by unknown” should be used as default value (<device_name> is a Characteristic of GAP)	This characteristic represents the name of the manufacturer of the device.	N
Manufacturer Details Link (URL)	mnml	URL to manufacturer (not to exceed 32 characters)	N	(none)	(none)	
Model Number	mnmo	Model number as designated by manufacturer	N	Model Number String (Device Information)	This characteristic represents the model number that is assigned by the device vendor.	N
Date of Manufacture	mndt	Manufacturing date of device	N	(none)	(none)	
Platform Version	mnpv	Version of platform – string (defined by manufacturer)	N	Software Revision String (Device Information)	This characteristic represents the software revision for the software within the device.	N
OS Version	mnos	Version of platform resident OS – string (defined by manufacturer)	N	(none)	BLE device usually has no os.	

Hardware Version	mnhw	Version of platform hardware	N	Hardware Revision String (Device Information)	This characteristic represents the hardware revision for the hardware within the device.	N
Firmware version	mnfv	Version of device firmware	N	Firmware Revision String (Device Information)	This characteristic represents the firmware revision for the firmware within the device.	N
Support URL	mnsl	URL that points to support information from manufacturer	N	(none)	(none)	
SystemTime	st	Reference time for the device	N	(none)	(none)	
Vendor ID	vid	Vendor defined string for the platform. The string is freeform and up to the vendor on what text to populate it.	N	Manufacturer Name String (Device Information)	This characteristic represents the name of the manufacturer of the device.	N

Table 7 shows how configurable OCF Platform properties, as specified in Table 16 in OCF Core Specification, should be derived as follows, if a BLE device does not implement Device Information Service, oic.wk.con.p should not be created on the Virtual OCF Server:

Table 7: oic.wk.con.p Resource Type definition

To OCF Property title	OCF Property name	OCF Description	OCF Mandatory?	From BLE Device Service Characteristic value	BLE Description	BLE Mandatory?
Platform Names	mnpn	Platform Identifier	N	Manufacturer Name String (Device Information)	This characteristic represents the name of the manufacturer of the device.	

No BLE Service equivalence exist for factory reset or restart, so there is no Characteristics for oic.wk.mnt properties Factory_Reset and Reboot, so mapping for oic.wk.mnt is omitted.

8.2.2.2 On-the-fly Translation

If a BLE Profile is not in Table 3 (not belong to a well-defined set), a BLE Translator does not translate it (on-the-fly translation is not supported).

8.2.3 Protocol translation between BLE and OCF

GATT profile [Bluetooth Core Specification] describes not only Service/Characteristic data model but also Features how to manipulate it (see A.1). GATT Features define how GATT-based data exchanges

takes place. The GATT features are used when we translate OCF CRUDN into BLE protocol and vice versa.

Table 8 shows translation rule between BLE GATT Feature and OCF CRUDN. When a BLE Translator receives CREATE/DELETE request from OCF client, it shall return corresponding error (4.xx or 5.xx) because there are no corresponding Features for them. If a BLE Translator receives RETRIEVE/UPDATE request from OCF client, it shall translate it into Characteristic Value Read/Characteristic Value Write respectively. NOTIFY request from OCF client shall be translated into Characteristic Descriptor Value Write, and Characteristic Value Notification/Indication from BLE GATT Server shall be translated into NOTIFICATION response.

Table 8 Protocol translation rule between BLE and OCF

BLE GATT Feature	OCF CRUDN
N/A (Not Supported)	CREATE
Characteristic Value Read	RETRIEVE
Characteristic Value Write	UPDATE
N/A (Not Supported)	DELETE
Characteristic Descriptor Value Write	NOTIFY request
Characteristic Value Notification/Indication	NOTIFICATION response

8.2.3.1 Initialization

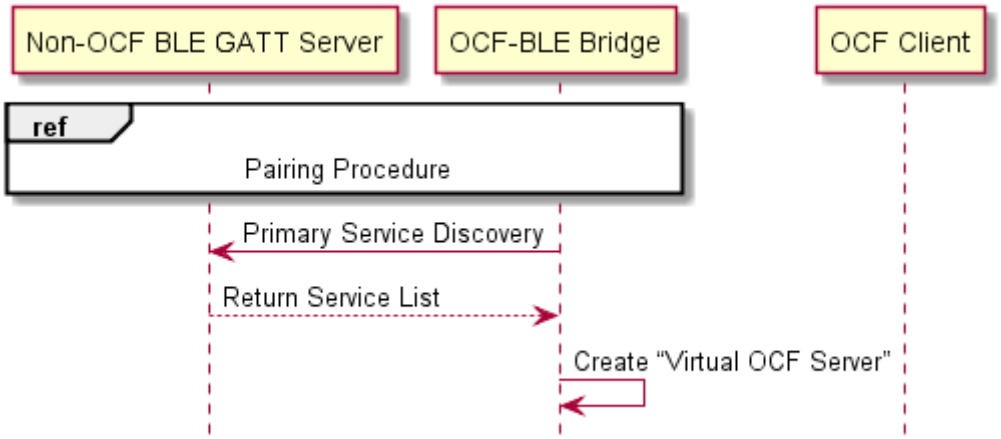
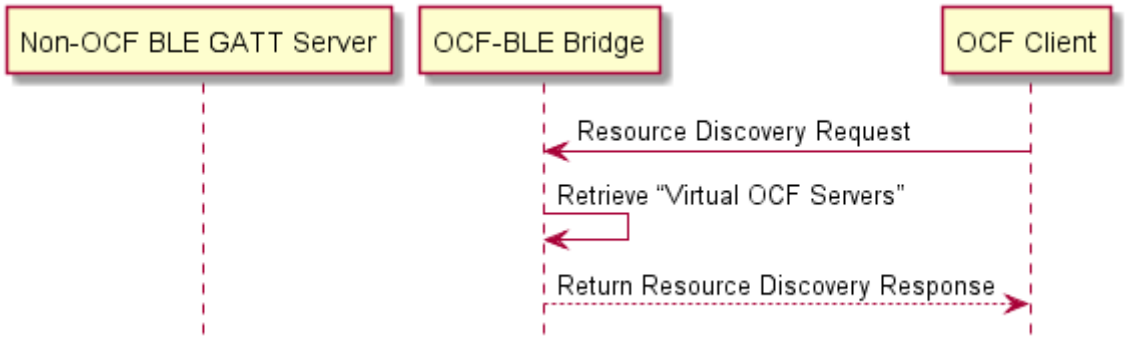


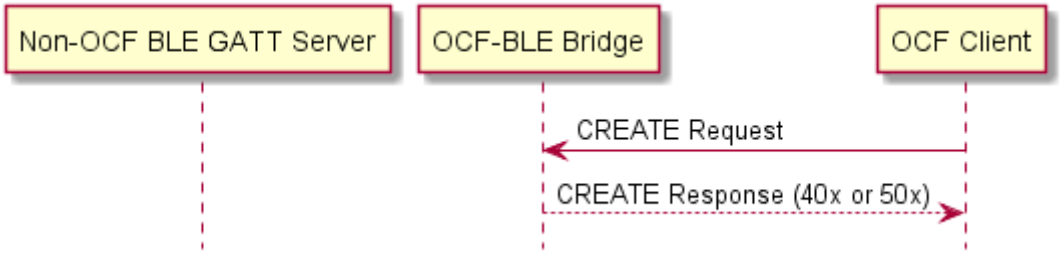
Figure 4 Initialization

252 **8.2.3.2 Resource Discovery**



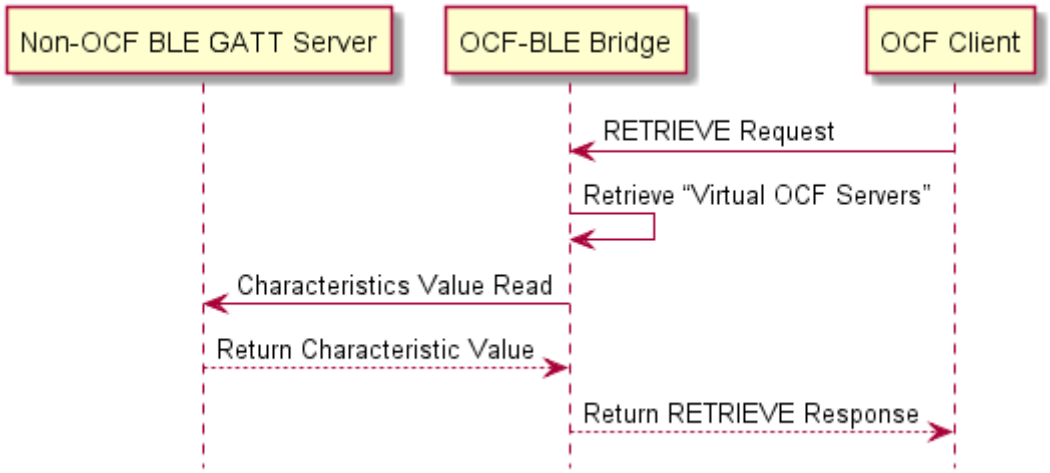
253
254 **Figure 5 Resource Discovery**

255 **8.2.3.3 Create Resource**



256
257 **Figure 6 Create Resource**

258 **8.2.3.4 Retrieve Resource**



259
260 **Figure 7 Retrieve Resource**

8.2.3.5 Update Resource

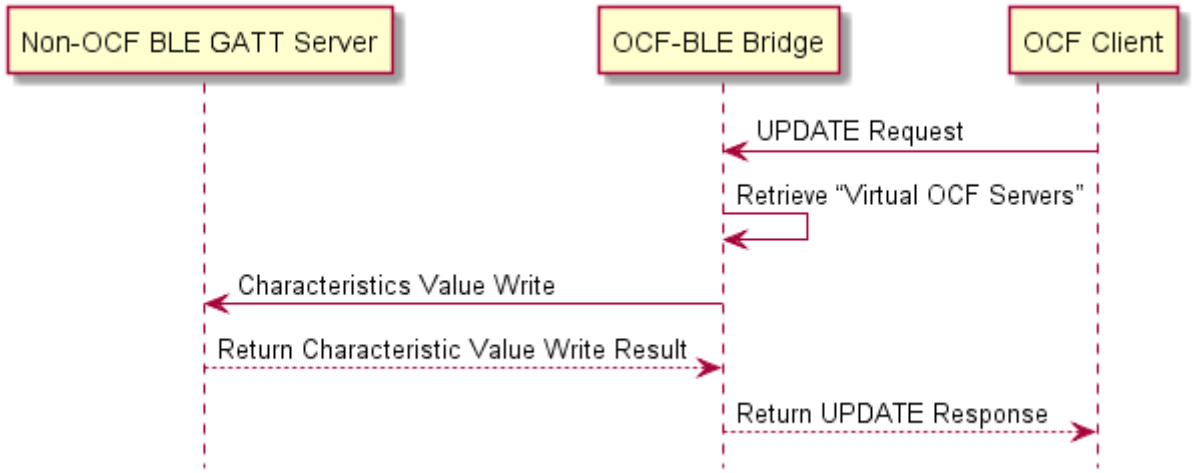


Figure 8 Update Resource

8.2.3.6 Delete Resource

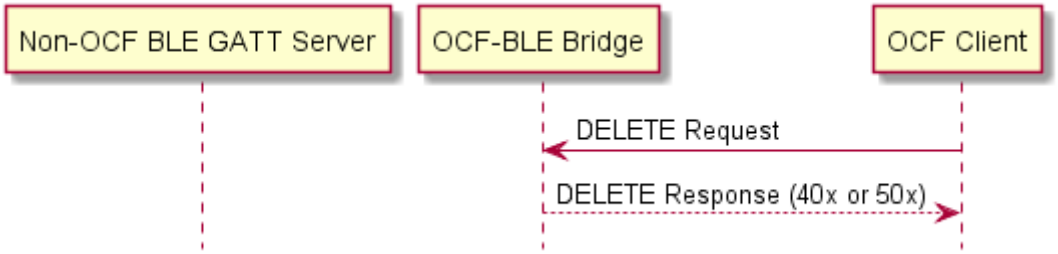


Figure 9 Delete Resource

8.2.3.7 Set Notification & Send Notification

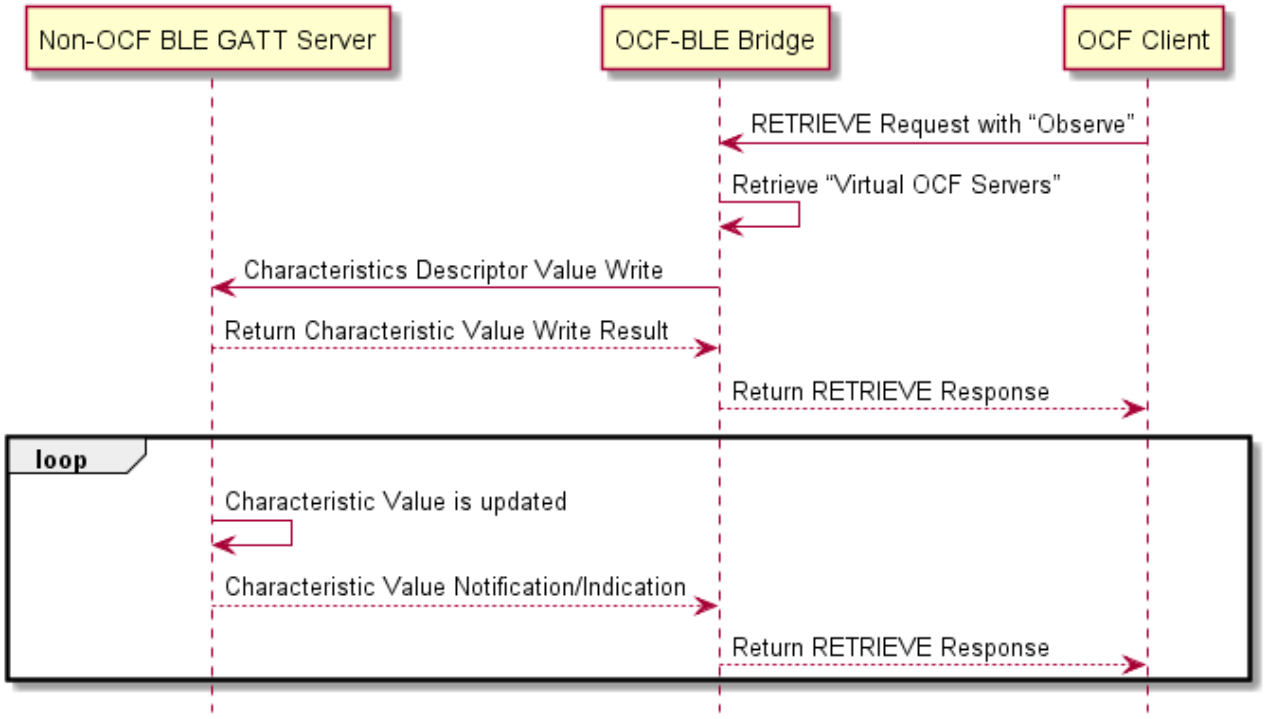


Figure 10 Set Notification & Send Notification

8.2.3.8 Error handling

If a BLE operation fails, the translator sends an appropriate OCF error response to the OCF Client. it constructs an appropriate OCF error message (e.g., diagnostic payload if using CoAP) from the BLE error name and error code (if any), using the form "<error name>: <error message>", with the <error name> taken from the ATT error code field and the <error message> taken from the ATT error name, and the error code for the OCF network set to an appropriate value.

8.2.4 Security

An OCF-BLE Bridge device shall satisfy section 5.7 Security of OCF Bridging Specification. An OCF Bridge Device shall block the communication of all OCF Devices with all Bridged Devices that don't communicate securely with the OCF Bridge Device.

12 Device Type definitions

No additional definitions are required.

13 Resource Type definitions

Since on-the-fly translation is out of scope, general Resource Type of Collection Resource for BLE devices, e.g., oic.r.bleobject is not required.

Annex A (Informative) BLE GATT based Data Model

A.1 BLE GATT based data model & GATT features

The Generic Attribute Profile (GATT) defines a service framework using the Attribute Protocol. This framework defines procedures and formats of services and their characteristics. The procedures defined include discovering, reading, writing, notifying and indicating characteristics, as well as configuring the broadcast of characteristics.

Profile dependency

Figure 11 depicts the structure and the dependencies of the profiles. A profile is dependent upon another profile if it re-uses parts of that profile by implicitly or explicitly referencing it.

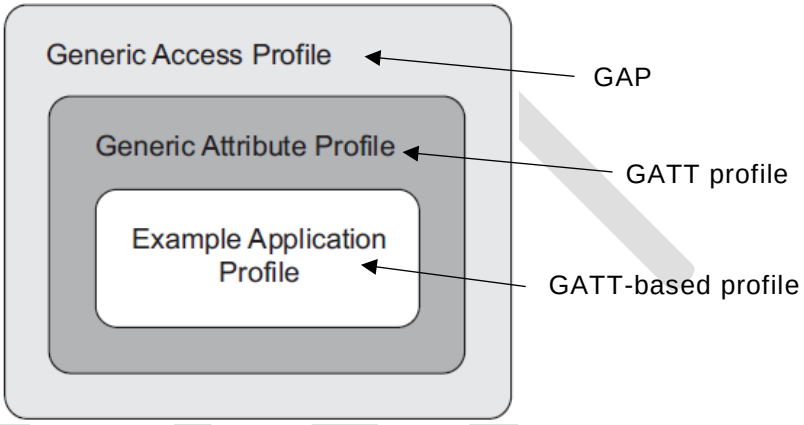


Figure 11 profile dependencies

Configurations and roles

There are two roles defined in GATT profile:

- Client: This is the device that initiates commands and requests towards the server and can receive responses, indications and notifications sent by the server.
- Server: This is the device that accepts incoming commands and requests from the client and sends responses, indications and notifications to a client.

A device can act in both roles at the same time.

GATT profile hierarchy

The GATT Profile specifies the structure in which profile data is exchanged. This structure defines basic elements such as services and characteristics, used in a profile. All of the elements are contained by Attributes. Attributes used in the ATT are containers that carry this profile data.

The top level of the hierarchy is a profile. A profile is composed of one or more services necessary to fulfil a use case. A service is composed of characteristics or references to other services. Each characteristic contains a value and may contain optional information about the value. The service and characteristic and the components of the characteristic (i.e. value and descriptors) contain the profile data and are all stored in Attributes on the server.

Under GATT profile, entity that provides Service-Characteristics data model plays “server” role and entity that gets data from GATT server plays “client” role.

There are other application profiles based on GATT profile. They are called “GATT-based profiles”.

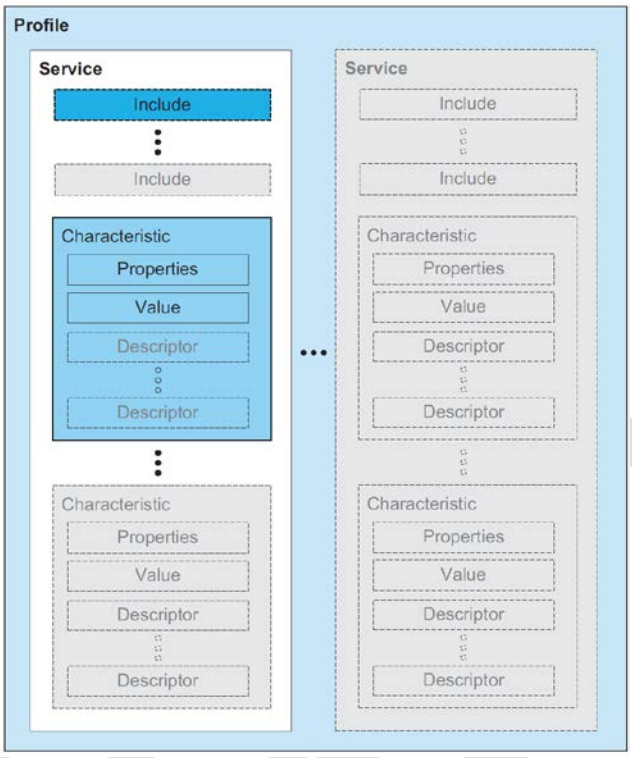


Figure 12 GATT profile hierarchy

Characteristic

A characteristic is a value used in a service along with properties and configuration information about how the value is accessed and information about how the value is displayed or represented. In GATT, a characteristic is defined by its characteristic definition. A characteristic definition contains a characteristic declaration, characteristic properties, and a value and may contain descriptors that describe the value or permit configuration of the server with respect to the characteristic.

GATT features

GATT profile also supports GATT features. GATT feature defines how GATT-based data exchanges take place. Each feature is mapped to one or more sub-procedures. These sub-procedures describe how the ATT is used to accomplish the corresponding feature.

Table 9 GATT Features & ATT protocol

	Feature	Sub-procedure	ATT protocol
1	Server Configuration	Exchange MTU	Exchange MTU Request Exchange MTU Response Error Response
2	Primary Service Discovery	Discover All Primary Services	Read By Group Type Request

			Read By Group Type Response Error Response
		Discover Primary Services by service UUID	Find By Type Value Request Find By Type Value Response Error Response
3	Relationship Discovery	Find Included Services	Read By Type Request Read By Type Response Error Response
4	Characteristic Discovery	Discover All Characteristic of a Service	Read By Type Request Read By Type Response Error Response
		Discover Characteristic by UUID	Read By Type Request Read By Type Response Error Response
5	Characteristic Descriptor Discovery	Discover All Characteristic Descriptors	Find Information Request Find Information Response Error Response
6	Characteristic Value Read	Read Characteristic Value	Read Request Read Response Error Response
		Read Using Characteristic UUID	Read By Type Request Read By Type Response Error Response
		Read Long Characteristic Values	Read Blob Request Read Blob Response Error Response
		Read Multiple Characteristic Values	Read Multiple Request Read Multiple Response Error Response
7	Characteristic Value Write	Write Without Response	Write Command
		Signed Write Without Response	Write Command
		Write Characteristic Value	Write Request Write Response Error Response
		Write Long Characteristic Values	Prepare Write Request Prepare Write Response Execute Write Request Execute Write Response Error Response
		Characteristic Value Reliable Writes	Prepare Write Request Prepare Write Response Execute Write Request Execute Write Response Error Response
8	Characteristic Value Notification	Notifications	Handle Value Notification
9	Characteristic Value Indication	Indications	Handle Value Indication Handle Value Confirmation

10	Characteristic Descriptor Value Read	Read Characteristic Descriptors	Read Request Read Response Error Response
		Read Long Characteristic Descriptors	Read Blob Request Read Blob Response Error Response
11	Characteristic Descriptor Value Write	Write Characteristic Descriptors	Write Request Write Response Error Response
		Write Long Characteristic Descriptors	Prepare Write Request Prepare Write Response Prepare Write Request Prepare Write Response Error Response

329

330

331

332

Annex B (Informative) Supporting Atomic Measurement Operation in BLE

B.1 Atomic Measurement Resource Type in OCF

Most OCF healthcare devices adopts Atomic Measurement feature. Atomic Measurement Resource Type is a specialisation of a Collection to ensure that the Client can only access the Properties of the linked Resources as a single group. Thus, if an OCF device corresponding to a BLE device implements Atomic Measurement Resource Type, the BLE Translator should guarantee that BLE GATT Characteristic values corresponding to properties of the Atomic Measurement Resource Type can be retrieved in atomic way.

B.2 Case 1. One Characteristic covers all properties of an Atomic Measurement Resource Type

In OCF-BLE mapping, a Service can be mapped to multiple OCF Resources and “a Characteristic” in a Service can be mapped to Properties in multiple OCF Resources. In general “Value of a Characteristic” is a byte stream (see below figure, below byte stream is a value of “blood pressure measurement Characteristic”). Usually “value of a Characteristic” includes multiple fields like below example and each field can be mapped to a property of OCF Resource.

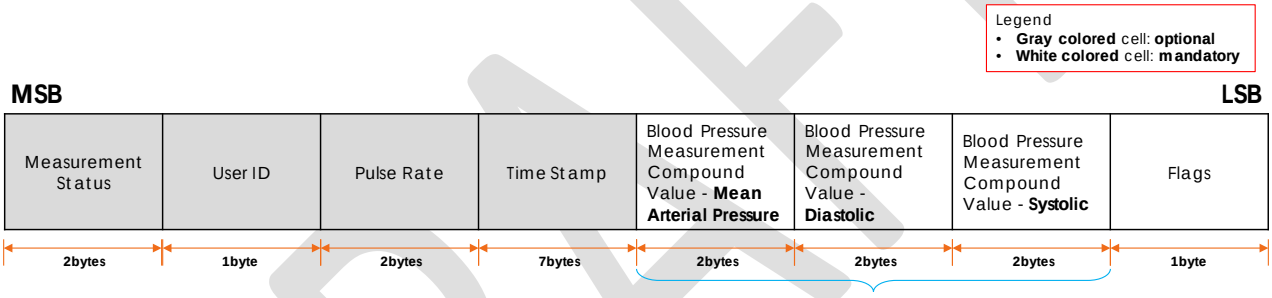


Figure 13 Value of blood pressure measurement Characteristic

For blood pressure device, “blood pressure measurement Characteristic” can cover all properties in bloodpressuremonitor-am. So if BLE GATT client (OCF-BLE Bridge device) uses “Read Characteristic Value” operation, it can get all values corresponding to all properties in bloodpressuremonitor-am at one time (atomic operation).

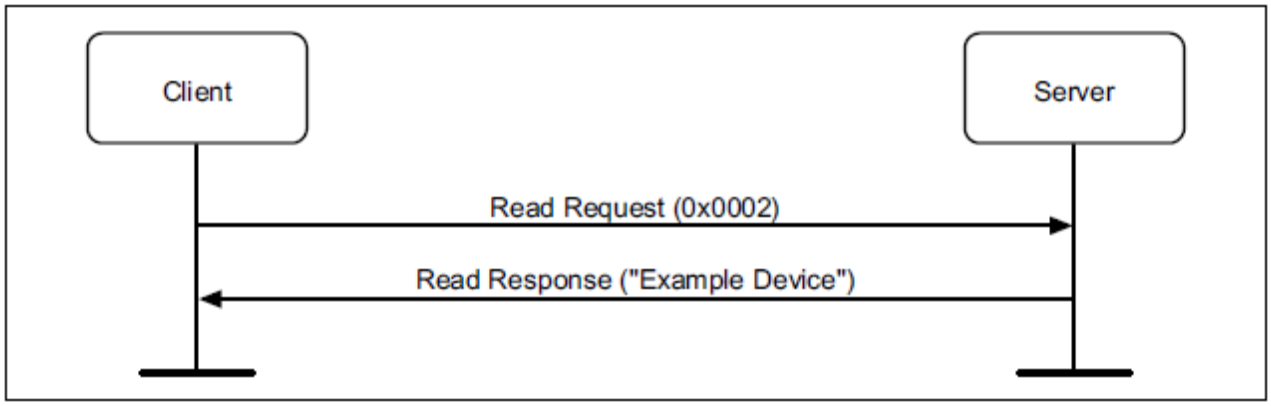


Figure 4.8: Read Characteristic Value example

B.3 Case 2. Multiple Characteristics cover all properties of an Atomic Measurement Resource Type

For glucose meter, 2 Characteristics (glucose measurement Characteristic, glucose measurement context Characteristic) cover all properties in glucosemeter-am. In this case, a BLE GATT client (OCF-BLE Bridge device) can use “Read Multiple Characteristic Values” operation to get multiple Characteristic values at one time.

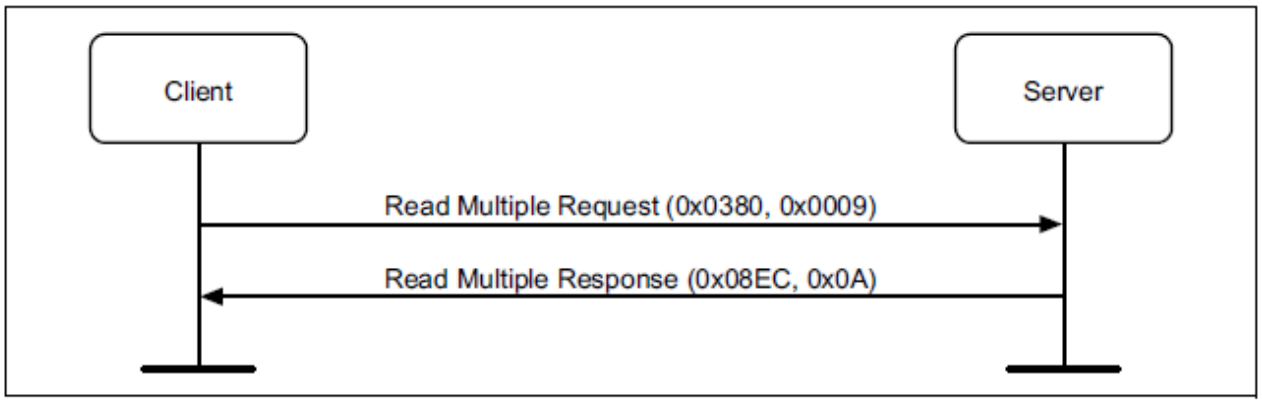


Figure 4.11: Read Multiple Characteristic Values example

However, some BLE GATT server may not support all operations except for “Notification”. In this case, a Characteristic value includes “sequence number” field, so BLE GATT client (OCF-BLE Bridge device) can make a set of values which are measured at the same time by using it.

Below examples are 2 Characteristics of glucose Service.

MSB LSB

Sensor Status Annunciation	Sample Location	Type	Glucose Concentration (kg/L or mol/L)	Time Offset	Base Time	Sequence Number	Flags
2 bytes	4 bits	4 bits	2 bytes	2 bytes	7 bytes	2 bytes	1 byte

Figure 14 Value of glucose measurement Characteristic

MSB LSB

HbA1c	Medication (kilograms or liter)	Medication ID	Exercise Intensity	Exercise Duration	Health	Tester	Meal	Carbohydrate (kilograms)	Carbohydrate ID	Extended Flags	Sequence Number	Flags
2 bytes	2 bytes	1 byte	1 byte	2 bytes	4 bits	4 bits	1 byte	2 bytes	1 bytes	1 bytes	2 bytes	1 byte

Figure 15 Value of glucose measurement context Characteristic

Annex C (Informative) BLE Security

C.1 Security Mode

BLE GAP supports two security modes, security mode 1 and security mode 2. Each security mode has several security levels. Further details can be found in Bluetooth Core Specification.

Table 10 GAP security mode

GAP security mode	security level
Security mode 1	1 (no security)
	2 (Unauthenticated pairing with encryption)
	3 (Authenticated pairing with encryption)
	4 (Authenticated LE Secure Connections pairing with encryption)
Security mode 2	1 (Unauthenticated pairing with data signing)
	2 (Authenticated pairing with data signing)

Security mode 1 and Security level 2 or higher would typically be considered secure from an OCF perspective. The appropriate selection security mode and level is left to the vendor.

Figure 16 shows how communications in both ecosystems of OCF-BLE Bridge device are secured by their own security.

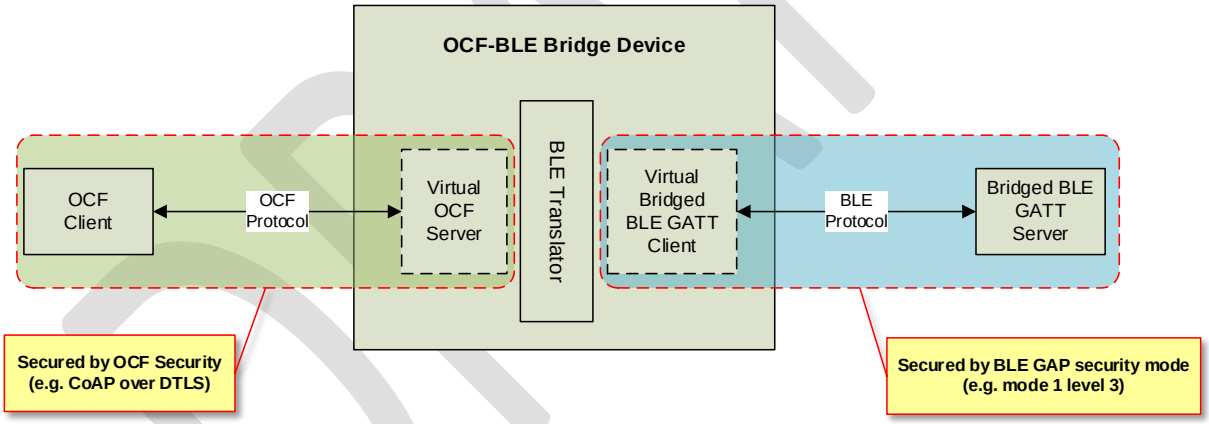


Figure 16 Security in OCF-BLE Bridge