WHITE PAPER

Smart Commercial Buildings Work Group

OPEN CONNECTIVITY FOUNDATION®

BACnet & OCF Core Framework

USING BACNET OBJECTSON OCF

March 5, 2020

Introduction

OCF devices can be used as a subordinate to a BACnet gateway. However, using OCF devices means the introduction of a new kind of device into the commercial building market, which needs application level translation by an BACnet gateway. The appealing part of OCF is the secure OCF Core Framework and that the framework can run on a Thread-based device. To avoid translating the OCF models into BACnet objects, the OCF Framework can also be used to transport the BACnet objects. This reduces the application knowledge, from BACnet perspective the application logic will be the same. Furthermore, there is no need to do additional work to perform a translation from OCF models into BACnet objects in a BACnet gateway.

BACnet Gateway with OCF device support

The OCF Core Framework has been designed to securely transport data between clients and servers. In this document a description is given of how to use the OCF Core Framework to transport BACnet objects. The BACnet specification defines different encoding schemes for BACnet objects for example the encoding schemes: (binary) ASN.1, JSON and XML are defined. The most deployed encoding is ASN.1, this encoding is also used in conjunction with the transport MS/TP. The JSON data formats (Annex Z) are more linked to the BACnet/WS, which is not widely deployed.

The encoding schemes are conveying the same models as defined per BACnet object. The only difference is the encoding used on the transport. The BACnet gateway does not have to create a translation since the content is the same, only the difference in encoding scheme has to be taken into account. The OCF Core Framework will be responsible for the communication between the devices and clients in a secure way. The actual data transported can be binary, for example it is either CBOR encoded JSON or ASN.1. Since OCF has been built on IPv6, different IP physical layers can be used. OCF currently supports wired ethernet, Wi-Fi and Thread, the meshed network defined by the Thread Group.

The resulting protocol stack combining the technologies is depicted in the figure below.

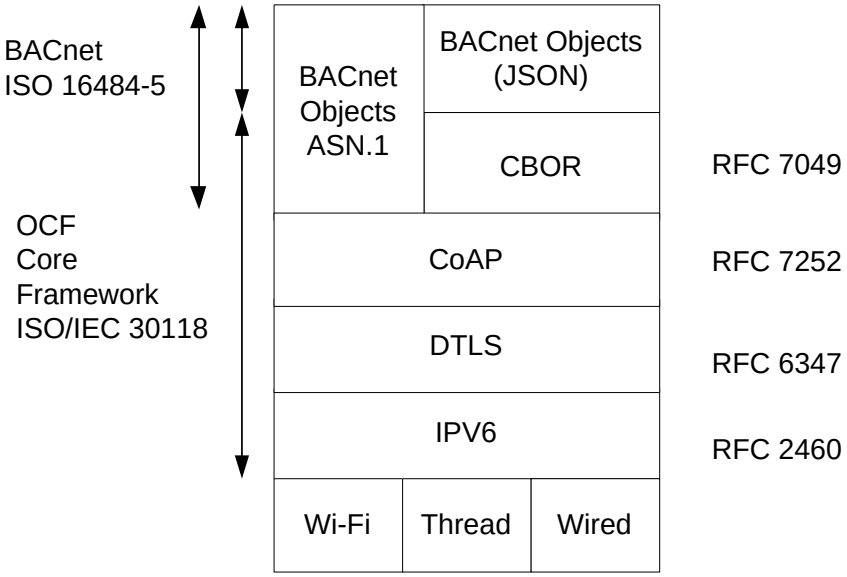

Figure 1 OCF Core Framework transporting BACnet Objects over different IP transports

The OCF Core Framework can carry all BACnet objects and does not use the BACnet addressing scheme but uses addressing based on IPV6. Using the IP addressing for devices allows easy integration of the protocol by means of a transport protocol gateway. The applications created on top of the BACnet will still work without conversion. E.g. only the transport of the BACnet objects is different.

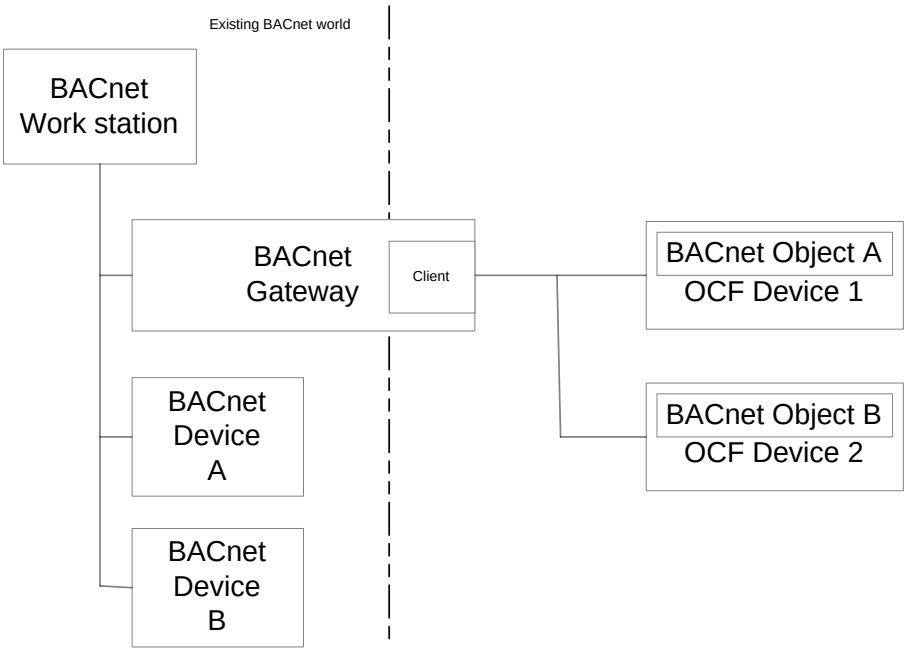

Figure 2 OCF integration into BACnet using a transport gateway

The BACnet gateway represents the OCF devices in the BACnet world as BACnet devices. The BACnet gateway will include an OCF client. The integrated OCF client will be able to discover the OCF devices. The information of the discovery process can be used to simulate the BACnet devices in the Gateway. Typical information that will be known through OCF discovery process is:

- The number of OCF devices
- Per OCF device
    - The implemented resources
        - Per resource
            - The interface to use to communicate with the resource

the resource type of the resource, e.g. identification of what the resource represents. The discovery information can be used to simulate BACnet devices and the BACnet objects.

The integrated OCF Client will be able to invoke methods on the implemented resources. The returned data on a GET method will be CBOR, however this will be automatically translated into JSON by an implementation. The data to perform an update action on an OCF device also can be created in JSON and will be automatically translated by an implementation. Therefore, using the BACnet objects with JSON encoding is a perfect fit to transport BACnet objects over the

OCF Framework. Furthermore, the interaction model of BACnet/WS can be used, e.g. replacing HTTP with CoAP will give the same interaction model as already defined by BACnet/WS.

Mapping of BACnet Object on to the OCF Resource model

The OCF resources can be recognized in the OCF eco system. This achieved via OCF discovery information. The discovery response specifies the instantiated resources on the OCF device, the response format defines an URL and link target attributes ("rt", "if") for the instantiated resources. The link target attribute resource type ("rt") defines the payload regarding the function of the resource. This mechanism can be used to indicate which BACnet Object definitions is instantiated at the resource destination. See section Y.4.1.1 of the BACnet specification for the naming convention of the BACnet Object Type definitions. The BACnet Object Type definition names start with "0-", for example the analog input object is denoted as "0-AnalogInputObject". To adhere to the OCF core specification the BACnet object models should be prefixed with "x-bacnet.org-". Hence the full resource type name of the analog input object will be "x-bacnet.org-0-AnalogInputObject".

The link target attribute interface ("if") signals which CoAP verbs (GET, POST, PUT, DELETE) are implemented on the URL. For example, an URL with rt="oic.r.temperature", if="oic.if.s" is a temperature sensor. The "if" interface indicates that the content can be read (CoAP verb GET). For an actuator the value "oic.if.a" can be used, this value indicates that additional to read the value also can be updated (CoAP verb POST). The 0-AnalogInputObject object type itself defines the payload to be used on the verb, e.g. wat to return on a read method and what can be updated on a write method. Since most BACnet objects have updatable parameters, the generic read/write ("oic.if.rw") interface value can be used for most BACnet objects.

The definition of the BACnet object identifies which parts are returned on a GET and a POST, Typically:

- The payload of an BACnet object will be CBOR encoded JSON.
    - The JSON Data formats of the BACnet objects is described in Annex Z of the BACnet standard.
- All implemented readable values should be returned on the GET method.

At least one of the writeable values should be used in the request payload of the POST method.

References

BACnet	ISO 16484-5:2017
Open Connectivity Foundation (OCF)	ISO/IEC 30118
Concise Binary Object Representation (CBOR)	https://tools.ietf.org/html/rfc7049
The Constrained Application Protocol (CoAP)	https://tools.ietf.org/html/rfc7252
Datagram Transport Layer Security	https://tools.ietf.org/html/rfc6347
Standardized rt values:	https://www.iana.org/assignments/core-parameters/core-parameters.xhtml#rt-link-target-att-value
Thread Group	https://www.threadgroup.org/