

AllJoyn Device System Bridge

Connecting device ecosystems

Abstract

The document describes how different types of industrial and consumer devices can be integrated into the AllJoyn ecosystem. With the Device System Bridge, Microsoft enables developers and system integrators to easily create scenarios with a wide range of different devices.

Author(s)

Torsten Stein, Sr. Program Manager, Microsoft Internet of Things

Publication date

April, 2015

Version

1.0

We welcome your feedback on this paper. Please send your comments to

AllJoynInfo@microsoft.com.

Contributors

Javid Dhankwala, Nicolas Guibourge, Stuart Schaefer, Jon Slobodzian, Artem Zhurid

Reviewers

Jason Farmer, Erdinc Korpeoglu, Soemin Tjong and Blake T Walsh

The information contained in this document represents the current view of Microsoft Corporation on the issues discussed as of the date of publication. Because Microsoft must respond to changing market conditions, it should not be interpreted to be a commitment on the part of Microsoft, and Microsoft cannot guarantee the accuracy of any information presented after the date of publication.

MICROSOFT MAKES NO WARRANTIES, EXPRESS, IMPLIED OR STATUTORY, AS TO THE INFORMATION IN THIS DOCUMENT.

Complying with all applicable copyright laws is the responsibility of the user. Without limiting the rights under copyright, no part of this document may be reproduced, stored in or introduced into a retrieval system, or transmitted in any form or by any means (electronic, mechanical, photocopying, recording, or otherwise), or for any purpose, without the express written permission of Microsoft Corporation.

Microsoft may have patents, patent applications, trademarks, copyrights, or other intellectual property rights covering subject matter in this document. Except as expressly provided in any written license agreement from Microsoft, the furnishing of this document does not give you any license to these patents, trademarks, copyrights, or other intellectual property.

The descriptions of other companies' products in this document, if any, are provided only as a convenience to you. Any such references should not be considered an endorsement or support by Microsoft. Microsoft cannot guarantee their accuracy, and the products may change over time. Also, the descriptions are intended as brief highlights to aid understanding, rather than as thorough coverage. For authoritative descriptions of these products, please consult their respective manufacturers.

© 2015 Microsoft Corporation. All rights reserved. Any use or distribution of these materials without express authorization of Microsoft Corp. is strictly prohibited.

Microsoft and Windows are either registered trademarks of Microsoft Corporation in the United States and/or other countries.

The names of actual companies and products mentioned herein may be the trademarks of their respective owners.

Table of contents

1	Executive summary	1
2	DSB – AllJoyn for everything	2
2.1	Device System Bridge vs. AllJoyn Gateway	3
2.2	Device System Bridge vs. AllJoyn transport	3
2.3	Design overview	3
2.4	Configuration	5
2.4.1	<i>Bridge configuration</i>	6
2.4.2	<i>Adapter configuration sample</i>	8
2.5	Scalability	8
2.6	Development	9
3	References	10

1 Executive summary

AllJoyn is a fast growing open source project to facilitate interoperability between devices. It enables devices to share capabilities, advertise their presence and communicate with each other on a local network. The system is designed to work across, and independently from, device connection technologies (e.g. Wi-Fi or Bluetooth), platforms (Windows, Linux ...) and manufacturers. With this, users can choose from a wide variety of devices and can be assured that these devices work with each other to enable great experiences.

“...My TV shows that someone is at the front door and I can turn my lights on with my smart watch ... it just works.”

Microsoft sees the rapid growth of the AllSeen Alliance¹ as an indicator for the growth of the ecosystem and industry wide adoption of AllJoyn.

As new technologies such as AllJoyn bring new opportunities, they often leave companies and users with tough financial decisions on how to manage the TCO² of their investments. In order to realize the full capabilities of AllJoyn they might need to replace their existing device with devices that support AllJoyn or update their existing devices with AllJoyn-capable firmware. In many cases, these options are not feasible due to device cost, technical limitations, or AllJoyn firmware availability for the device.

In order to address these challenges and broaden the reach of AllJoyn, Microsoft developed the concept of a Device System Bridge (henceforth DSB). The DSB enables non-AllJoyn devices to be included in the AllJoyn ecosystem. A DSB uses existing device interfaces to access the non-AllJoyn devices and creates a virtual proxy of these devices on the AllJoyn bus. With DSB enabled, AllJoyn devices and non-AllJoyn devices can now interact seamlessly.

¹ AllSeen Alliance is the collaboration project to drive the development of technologies around the Internet of Everything including AllJoyn: <http://allseenalliance.org>

² TCO: Total Cost of Ownership

2 DSB – AllJoyn for everything

The primary targets for the DSB are purpose-built or pre-existing device networks. The DSB allows a cost efficient and quick method to bring the capabilities of the devices in these networks to AllJoyn and the benefits of AllJoyn to the entire network without the need to add AllJoyn capabilities to device itself. Using smart home as an example, Figure 1 shows a typical mixture of devices used at home. With AllJoyn and the support of DSBs a user can have great experiences driven by the interoperability of all of those devices.

- A Z-Wave doorbell initiates a pop-up message on the AllJoyn enabled TV announcing that someone is at the door.
- With a TV remote a user can open the Z-Wave door lock
- An AllJoyn enabled light bulb can be turned on via a Bluetooth Smart watch

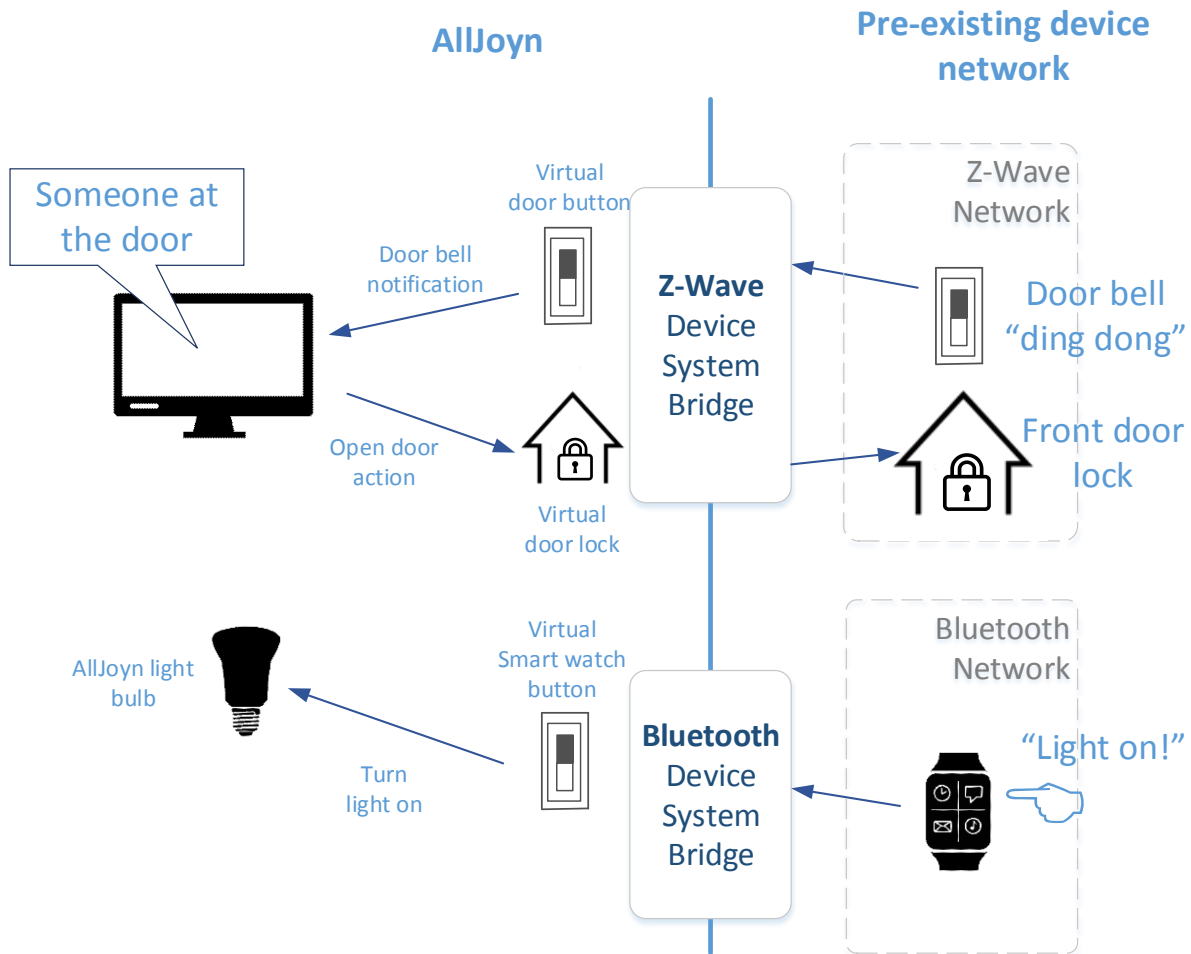


Figure 1 Smart Home scenario

2.1 Device System Bridge vs. AllJoyn Gateway

The AllSeen Alliance has specified an [AllJoyn Gateway Agent](#). When considering the Device System Bridge it is important to understand how it is different, and what scenarios are best suited to each AllJoyn component.

The key difference between the DSB and the AllJoyn Gateway Agent is that the AllJoyn Gateway Agent enables remote (outside of the proximal network) control of AllJoyn devices through one or multiple external networks. In contrast, the Device System Bridge creates a virtual proxy for non-AllJoyn devices on the AllJoyn bus. These devices can be local or remote depending on the design of the DSB.

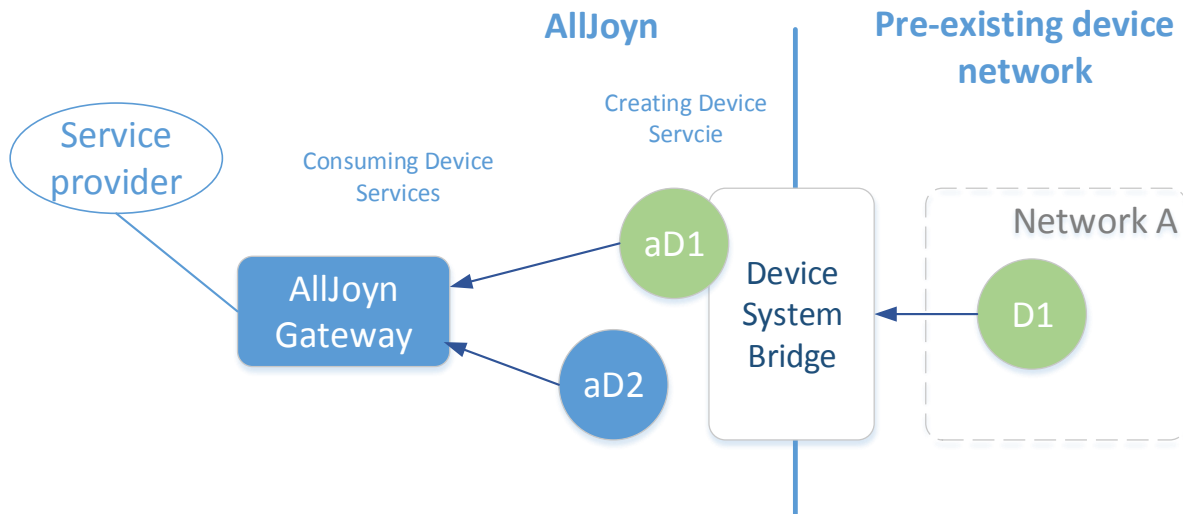


Figure 2 AllJoyn Gateway and DSB

Applying both components in the ecosystem allows comprehensive scenarios for device interaction and control.

2.2 Device System Bridge vs. AllJoyn transport

Device System Bridge is not an alternative to AllJoyn transports. When AllJoyn supports Bluetooth as a transport, the DSB can still play a role for pre-existing Bluetooth devices which are not AllJoyn capable.

When connecting via the Bluetooth transport, a Bluetooth device will have to implement the AllJoyn framework, at least the thin client, as a Bluetooth Profile on the device. When proxied by a DSB, the DSB will provide the AllJoyn framework and the interconnection to whatever interface (e.g. one of the Bluetooth Profiles) the device natively provides.

2.3 Design overview

Each DSB is implemented as a separate AllJoyn application. Multiple DSBs can be connected to the same AllJoyn bus. Figure 3 shows how devices from purpose built networks will appear on an AllJoyn network. In this example there are two different systems each having their own Device System Bridge, DSB A and DSB B. On the AllJoyn side the devices from these networks D1 ... D5 will appear as AllJoyn bus

attachments aD1 ... aD5, enabling interactions with each other and other AllJoyn devices (aD6 ... aD8). On the non-AllJoyn enabled side the device D1... D5 will not be able to discover the AllJoyn devices as there is no reverse proxy for aD6...aD8 on the alternative network. This network will remain closed to its own participating devices unless the author provides additional functionality beyond what the DSB framework directly enables. DSB does not address this problem due to the wide variety and requirements of these other device systems.

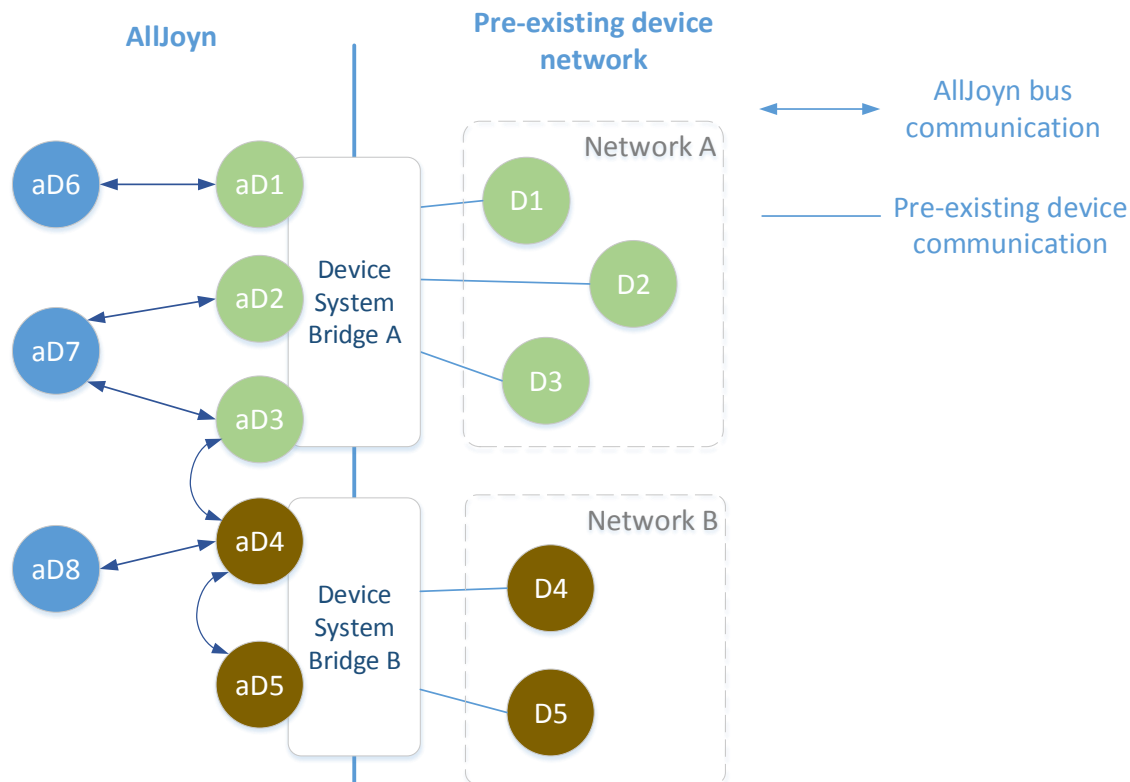


Figure 3 Device mapping through System Bridges.

Windows 10 provides an AllJoyn router node as part of the OS. The DSB depends on these OS supplied AllJoyn APIs and the AllJoyn Framework for enablement. In Windows, the DSB is designed as a single Universal Application for each DSB type, e.g. Z-Wave or BACnet. This takes advantage of the application isolation model providing OS level separation of DSBs running on the same machine.

A DSB has three structural components:

Communication Stack or Interface provides interconnection to the purpose built device system.

Adapter instantiates and manages a virtual device on behalf of each device from the alternative network that can be exposed to the AllJoyn bus. This information is consumed by the Bridge.

Bridge instantiates a bus attachment for each of these devices. The bridge also exposes an AllJoyn bus attachment for itself with three bus objects: one which implements the AllJoyn

standard About interface and two configuration interfaces³, one for the Bridge and one for the Adapter. The configuration interfaces provide settings to the DSB including:

- Access information for connecting to the alternate network and its devices (e.g. server name or access PIN),
- List of devices to be exposed as AllJoyn bus object
- Security settings for each bus object. More information can be found in the security and configuration sections below.

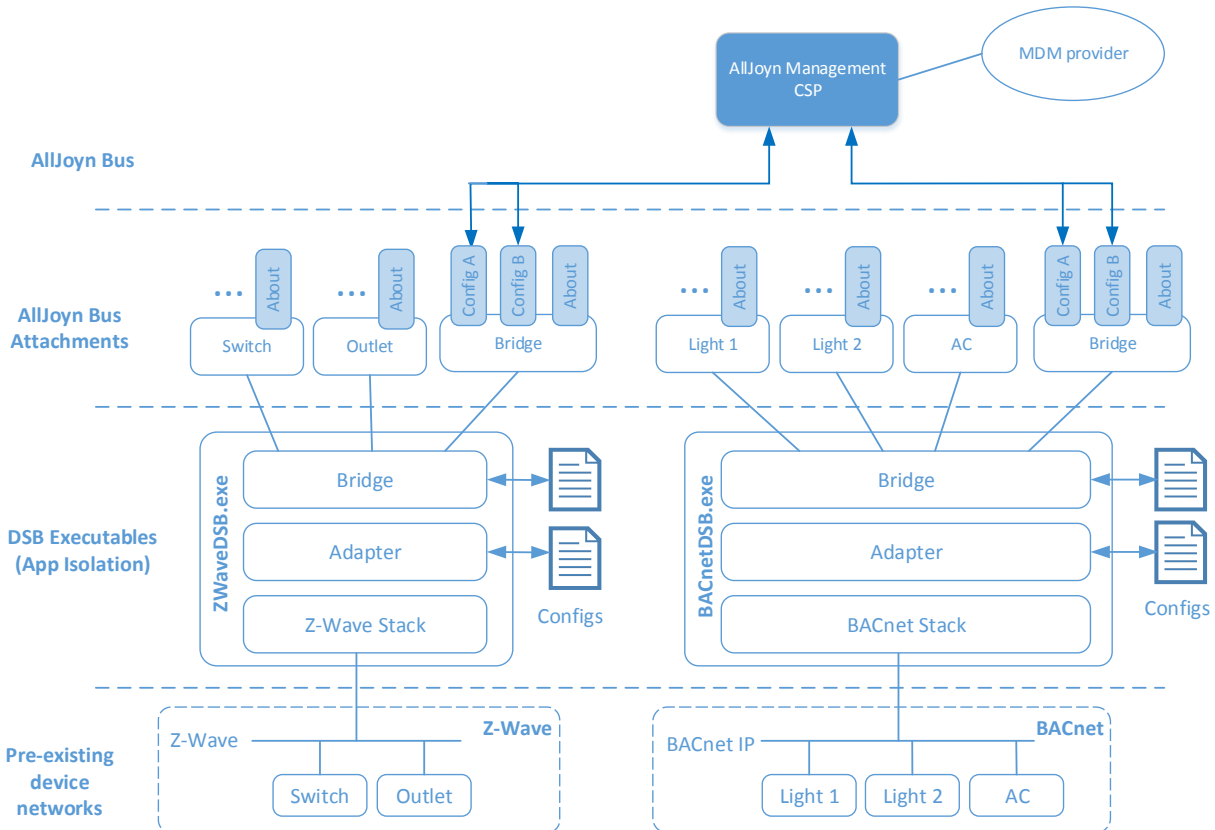


Figure 4 Device System Bridge overview

The DSB currently supports [AllJoyn Security 1.0](#) for all Bus Attachments. Security settings are configured through the bridge configuration interface.

2.4 Configuration

Configuration of the DSB is a key feature of the design. There are two ways of providing the information to the DSB, locally by changing the configuration file or externally through the Microsoft configuration interface - `com.microsoft.alljoynmanagement.config`.

³ This is a proprietary configuration interface - not the AllJoyn Configuration Interface ([org.alljoyn.Config](#))

For enterprise applications Microsoft provides the AllJoyn Management CSP⁴ (see Figure 4) which allows enterprise management tools such as System Center or InTune to configure the available DSBs. The AllJoyn Management CSP will query bus attachments for the *com.microsoft.alljoynmanagement.config* interface.

There are two separate configuration interfaces exposed by a DSB one for the Bridge and one for the Adapter. The configuration structure for the Bridge interface is the same for each DSB containing the bridge and device configuration – see chapter 2.4.1. The structure of the Adapter is specific for each DSB containing e.g. access information for the pre-existing device network – see chapter 2.4.2.

Currently the DSB is not using the AllJoyn Configuration Interface as, in its current definition, it does not allow streaming of complex configuration files. In the future, this capability could be integrated in the AllJoyn Configuration Service.

2.4.1 Bridge configuration

The XML file below shows an example of a Bridge configuration for three devices, only one of which is exposed to the AllJoyn bus. When a DSB runs the first time, a default configuration file is generated that includes all discovered devices on the pre-existing device network.

```
<?xml version="1.0" encoding="utf-8"?>
<BridgeConfig>
  <Settings>
    <Bridge>
      <KEYX></KEYX>
    </Bridge>
    <Device>
      <DefaultVisibility>>false</DefaultVisibility>
      <KEYX></KEYX>
      <USERNAME></USERNAME>
      <PASSWORD></PASSWORD>
      <ECDHEECDSAPRIVATEKEY></ECDHEECDSAPRIVATEKEY>
      <ECDHEECDSACERTCHAIN></ECDHEECDSACERTCHAIN>
    </Device>
  </Settings>
  <Objects>
    <Object Id="001-001-001" Visible="false">
      <Desc>2 X Switch</Desc>
    </Object>
    <Object Id="001-002-001" Visible="true">
      <Desc>Dim Control 725</Desc>
    </Object>
    <Object Id="001-003-001" Visible="false">
      <Desc>Temperature Sensor 155</Desc>
    </Object>
  </Objects>
</BridgeConfig>
```

⁴ CSP: Configuration Service Provider,
[https://msdn.microsoft.com/en-us/library/windows/hardware/dn920025\(v=vs.85\).aspx](https://msdn.microsoft.com/en-us/library/windows/hardware/dn920025(v=vs.85).aspx)

```
</Objects>  
</BridgeConfig>
```

Note: The XML only shows authentication settings under the <Settings><Device> node for the Bridge. These will apply to all devices under the <Objects> node. In a later release this will be changed and the authentication settings will be configurable for each device separately.

Bridge\KEYX

DSB Bridge Configuration Authentication Key. When blank, the Bridge and Adapter Configuration files can be modified by any AllJoyn client. Any conforming ALLJOYN_SRP_KEYX value enables AllJoyn Authentication for the Bridge and Adapter Configuration files.

Device\DefaultVisibility

Sets whether or not devices found on the configured network are automatically visible to the AllJoyn network. If a new device appears when true, the device will automatically appear on the network. When false, the device will not appear on the AllJoyn network and must be manually configured.

Device\KEYX

Enables ALLJOYN_SRP_KEYX authentication mode for all AllJoyn exposed devices from the configured network. The key value must be an ALLJOYN_SRP_KEYX conforming values.

Device\USERNAME and Device\PASSWORD

Enables ALLJOYN_SRP_LOGON authentication mode for all AllJoyn exposed devices from the configured network. Username and Password must be ALLJOYN_SRP_LOGON conforming values.

Device\ECDHEECDSAPRIVATEKEY and Device\ECDHEECDSACERTCHAIN

Enables ALLJOYN_ECDHE_ECDSA authentication mode for all virtual devices exposed on the AllJoyn network. ECDHEECDSAPRIVATEKEY and ECDHEECDSACERTCHAIN must be ALLJOYN_ECDHE_ECDSA conforming values.

Object:Id

Unique ID of a specific virtual device (not configurable) as it appears on the AllJoyn Network. This is automatically generated when a device is discovered on the configured network.

Object:Visible

Current visibility of virtual device on the AllJoyn Network. During device discovery, this value is set to the **Device\DefaultVisibility** setting.

Object\Desc

Description of the virtual device, usually returned from the physical device itself or controllers on the configured purpose built network. During device discovery, this value is automatically populated. When the description is manually changed, it will not be overridden.

2.4.2 Adapter configuration sample

The sample XML below shows the configuration for a BACnet DSB adapter. The configuration details are Adapter specific and are defined by each adapter’s implementer.

```
<?xml version="1.0" encoding="UTF-8"?>
<BACnetConfig>
  <BACnetStack BBMD_IPAddress="xxx.xxx.xxx.xxx" BBMD_Port="yyyyy" NetworkInterface=""
RequestPriority="8" DeviceInstanceMin="-1" DeviceInstanceMax="-1"/>
  <AllowedDeviceList>
    <Allowed>device_model_filter_token</Allowed>
  </AllowedDeviceList>
</BACnetConfig>
```

2.5 Scalability

Multiple DSBs of the same kind can be connected to the same pre-existing device networks, while deployed on one AllJoyn bus for scalability. Each of the DSBs could potentially expose the same virtual devices and therefore create multiple instances on the AllJoyn bus for one device on the pre-existing device networks. In Figure 5 device D1 has two instances on the AllJoyn bus: aD1 and aD3.

One way to handle this issue with the current DSB design is to configure each DSB to only show a selected number of devices. This can be achieved through the bridge configuration interface – see chapter 2.4.1.

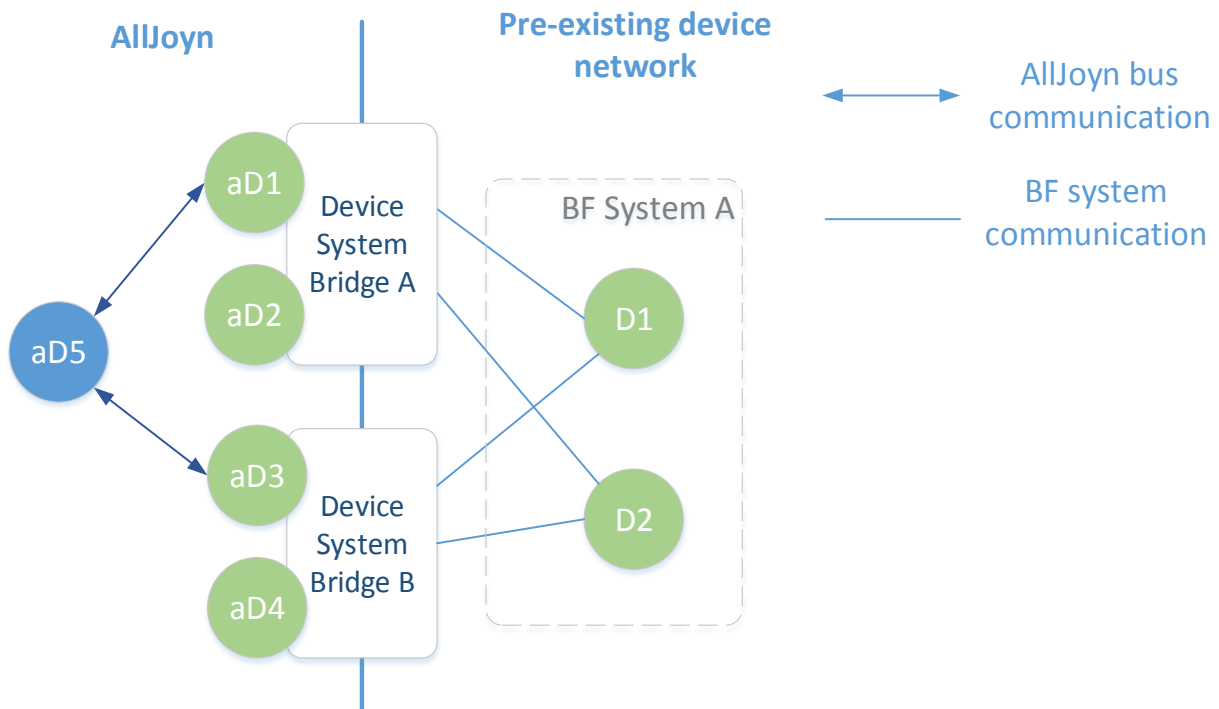


Figure 5 Virtual device duplication

2.6 Development

Microsoft will release a Visual Studio Project Template to ease DSB creation as well as several DSB samples. Developers will be able to find these through the Visual Studio online resources. In addition there will be DSB sample available as open source⁵:

- Mock DSB – simulating virtual devices
- Z-Wave DSB – connecting to a network of Z-Wave devices
- BACnet DSB – connecting to a BACnet system via BACnet IP

⁵ Target date //Build 2015

3 References

Link to Visual Studio Template – Search for “AllJoyn DSB”: <https://visualstudiogallery.msdn.microsoft.com>

Link to DSB samples Git: <https://git.allseenalliance.org/cgit/dsb.git/>

AllSeen Alliance working group: <https://wiki.allseenalliance.org/gateway/dsb>

AllJoyn Framework Architecture overview: <https://allseenalliance.org/developers/learn/architecture>