

Application Layer Interop White Paper

February 2018

This Thread Technical white paper is provided for reference purposes only. The full technical specification is available publicly. To join gain access, please follow this link: <https://www.threadgroup.org/ThreadSpec>.

If there are questions or comments on these technical papers, please send them to help@threadgroup.org.

This document and the information contained herein is provided on an “AS IS” basis and THE THREAD GROUP DISCLAIMS ALL WARRANTIES EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO (A) ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OF THIRD PARTIES (INCLUDING WITHOUT LIMITATION ANY INTELLECTUAL PROPERTY RIGHTS INCLUDING PATENT, COPYRIGHT OR TRADEMARK RIGHTS) OR (B) ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE OR NONINFRINGEMENT.

IN NO EVENT WILL THE THREAD GROUP BE LIABLE FOR ANY LOSS OF PROFITS, LOSS OF BUSINESS, LOSS OF USE OF DATA, INTERRUPTION OF BUSINESS, OR FOR ANY OTHER DIRECT, INDIRECT, SPECIAL OR EXEMPLARY, INCIDENTAL, PUNITIVE OR CONSEQUENTIAL DAMAGES OF ANY KIND, IN CONTRACT OR IN TORT, IN CONNECTION WITH THIS DOCUMENT OR THE INFORMATION CONTAINED HEREIN, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH LOSS OR DAMAGE.

Copyright © 2015 Thread Group, Inc. All rights reserved.

Application Layer Interop White Paper

February 2018

Revision History

Revision	Date	Comments
1.0	February 27 2018	Public Release

Table of Contents

I. Introduction

II. Interop Overview

III. Application Layer Standards

IV. Participants & Device Profiles

V. Physical Setup

VI. Interop Use Cases

A. Dotdot Use Cases

B. Lightweight M2M Use Cases

VII. Conclusion

I. Introduction

Thread is an IPv6-based networking protocol targeted for use in low-power, embedded consumer and commercial IoT (Internet of Things) devices. Thread's approach to wireless networking offers numerous technological advantages over other wireless standards, including a secure and reliable mesh network with no single point of failure, simple connectivity and low power. For a complete technical overview, please refer to [Thread Stack Fundamentals](#).

While Thread defines the essential properties of the network transport layer, including mesh communication, commissioning and network security, it does not set a standard for how specific device classes on the network should be queried and controlled; that is, the common high-level interfaces that devices should support in order to exercise their core functions. The definition and implementation of these interfaces falls to the Application Layer.

Standard application layers are essential for establishing an open (i.e. non-proprietary) and interoperable ecosystem of devices. Under ideal circumstances, a given device class supporting a standard application layer interface may be queried and controlled by another device supporting the same standard.

For example, an application layer standard may define the core functions of a thermostat as measurement of local temperature and control of heating and cooling set points. The application layer standard provides the data model and command protocol necessary for exercising these properties. As such, another system or device with knowledge of this interface has a standard method for interacting with the thermostat, without additional knowledge of the thermostat's origin or vendor-specific configuration.

II. Interop Overview

The long-term success of an open standard rests on the ability of different implementations to work with one another under real world conditions. Interoperation or "interop" events serve as a forum for developers to validate their implementations, find and resolve bugs and improve the underlying standard while demonstrating its efficacy for a wider audience. To these ends, the Thread Application Layer Interop was conducted during an open session at the Thread All Members Meeting in Vancouver, British Columbia on June 27, 2017.

The specific purpose of the interop was to demonstrate a variety of device classes from different vendors operating on the same Thread network. The point of entry into the network was standard Thread commissioning through multiple Border Routers. Each non-border router device supported a specific application layer. Devices supporting the same application layer standard were able to interoperate on the network while coexisting with devices supporting different application layer standards.


In addition to a variety of interop use cases, numerous development platforms and diagnostics platforms were described and demonstrated.



III. Application Layer Standards


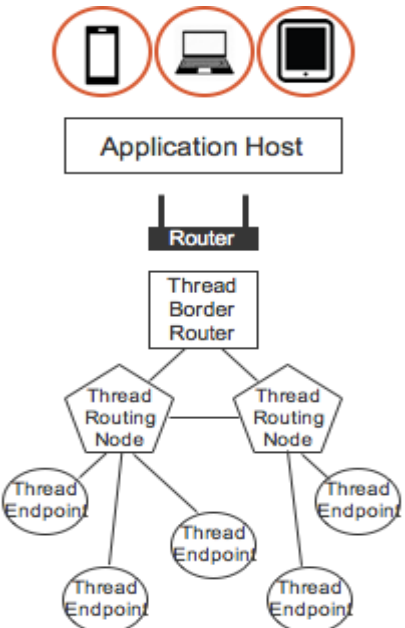
The application layer standards demonstrated included Dotdot by the Zigbee Alliance and Lightweight M2M by the Open Mobile Alliance.


IV. Participants & Device Profiles



The table below lists the vendors that participated in the interop as well as the systems, tools and devices they contributed:

Vendor	Device Profile
eero	<p>eero Generation 2 & eero Beacon</p>  <ul style="list-style-type: none"> • First open, agnostic Thread Border Router • Standard Thread commissioning via eero app • Whole-home WiFi and Thread coverage • NAT64, DNS64 and other services
Silicon Labs	Switch and Light

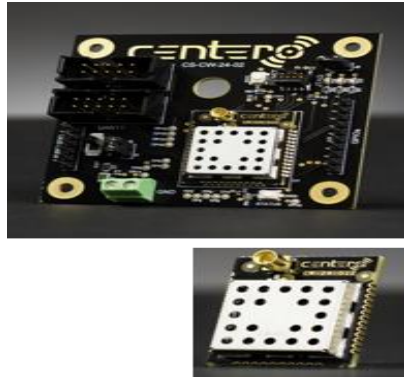
	 <p>The image shows two Silicon Labs Thread Node Application Development Systems and Platforms. The top device is a black PCB with a circular LED and various components. The bottom device is a white wall switch with the Silicon Labs logo.</p> <ul style="list-style-type: none"> • Supports Dotdot app layer • Supports standard Thread commissioning • Router (Mains) lighting device • Sleepy Device (Battery) light switch
<p>NXP</p>	<p>KW41Z Thread Node Application Development Systems and Platforms</p>  <p>The image shows the NXP KW41Z Thread Node Application Development Board, a green PCB with various components and a gold-colored antenna.</p> <ul style="list-style-type: none"> • Generic Thread application development platforms • “Arduino” form factor enabling different shields for lighting and sensing • Full featured certified NXP Thread Stack and OpenThread router or end device configurations • Supports Dotdot and LWM2M (with NXP Thread Stack)

	<p>Modular Gateway / Border Router Reference Design</p>  <p>The image shows the components of the Modular Gateway / Border Router Reference Design. It includes a white NXP V.JUNIOR1000 gateway device with an antenna, a small green PCB module, a SIM card, and two larger green PCB modules, one of which is an i.MX Application Processor.</p> <ul style="list-style-type: none"> • Thread Border Routers reference designs • Multi-radio modular design, including NFC for Thread commissioning • Plug-in module options for KW41Z (IEEE 802.15.5/BLE) • Embedded i.MX Application Processor run Embedded Linux i.MX 6UL/7Dual supports Android Things
<p>Arm</p>	<p>Arm mbed IoT Device Platform</p>  <p>The diagram illustrates the Arm mbed IoT Device Platform network architecture. At the top, three icons (smartphone, laptop, tablet) are grouped under an 'Application Host' box. Below this is a 'Router' box, which connects to a 'Thread Border Router' box. The Thread Border Router is connected to two 'Thread Routing Node' boxes. Each Thread Routing Node is connected to three 'Thread Endpoint' boxes, forming a mesh network structure.</p>

	<ul style="list-style-type: none"> • Complete support for Thread is built into mbed OS 5.4 • Cross-platform, multiple radio options • Border routers, routing nodes, low power endpoints • Mains and battery powered • Multiple app layers (demonstrating LWM2M) • Supports standard Thread commissioning
MMB Networks / ZEN Ecosystems	<p>MMB Mobile Application</p> <ul style="list-style-type: none"> • Supports Dotdot app layer • Supports control and query of Dotdot thermostat, door lock and lights <p>ZEN Thermostat</p>  <ul style="list-style-type: none"> • Supports Dotdot app layer • Supports standard Thread commissioning • Router (Mains) / Sleepy Device (Battery)
Yale (ASSA ABLOY)	Assure Lock

	 <ul style="list-style-type: none"> • Supports Dotdot app layer • Supports standard Thread commissioning • Sleepy Device
<p>Centero</p>	<p>SPiN Development Board</p>  <ul style="list-style-type: none"> • Supports LWM2M app layer (client) • Supports standard Thread commissioning • Line and Battery powered • Router/End Device/Sleepy End Device • LWM2M with IPSO Lighting Control + Temperature + Humidity Reference Designs • LWM2M with IPSO Distance/Range + Pressure Reference Designs

WEE Board




- Supports LWM2M app layer (client)
- Supports standard Thread commissioning
- Battery powered
- End Device/Sleepy End Device
- LWM2M with IPSO Lighting Control + Temperature + Humidity Reference Designs
- LWM2M with IPSO Distance/Range + Pressure Reference Designs

Wheel Border Router



- Supports LWM2M app layer (server)
- Supports standard Thread commissioning
- Line powered
- Border Router/Commissioner
- IPSO Lighting Control + Temperature + Humidity Reference Designs
- IPSO Distance/Range + Pressure Reference Designs

Ubilogix	<p data-bbox="505 302 867 338">Ubiqua Protocol Analyser</p>  <ul data-bbox="553 772 1409 1100" style="list-style-type: none"> • Supports Thread 1.1, Dotdot, CoAP, Zigbee 3.0, LWM2M • In-detail view of MLE, CoAP, 6LowPAN, DTLS, ICMP, DHCP, 802.15.4-2016 format fields inside a packet • View automatically decrypted MLE/MAC packets and reassembled 6LowPan fragmented packets • Visual representation of nodes in a network with graph and list structure • Real-time sniffing at both Main Network Station and Commissioning Station channel
----------	---

V. Physical Setup

In order to facilitate a wide-ranging demonstration, three “interop stations” were established during the interop event:

Main Network Station

This station facilitated the main multi-vendor application layer interop.

Tools & Diagnostics Station

This station displayed and demonstrated Thread development platforms, as well as diagnostics tools that tracked the actions and progress of the interop via network sniffing.

Commissioning Station

This station was provided as an open platform for testing standard Thread commissioning and was open to members not participating in the interop.

VI. Interop Use Cases

This section describes the application use-cases demonstrated during the interop.

A. Dotdot Use Cases

The eero border router was joined to the Ethernet LAN and served as the gateway to the Thread network.

The following Dotdot devices were commissioned on to the Thread network via the eero border router:

- ZEN Thermostat
- Yale Door Lock
- Silicon Labs LED Light & Switch
- NXP Development Platform for Dimmable Light and Controller

The MMB Mobile application operating on the LAN issued Dotdot commands to the devices on the network:

- ZEN Thermostat
 - Local Temperature query, Cool / Heat Set Point setting, Mode setting
- Yale Door Lock
 - Lock State query, Lock, Unlock
- Silicon Labs LED Light
 - On, Off

The Silicon Labs LED Light was also controlled via Dotdot commands (On/Off) by the Silicon Labs switch to which it was bound.

Additionally, the NXP Development Platform for Dotdot Controller operating on the Thread network issued On and Off commands to the NXP Development Platform for Dimmable Light, as well as commands and queries to the ZEN Thermostat and Yale Door Lock.

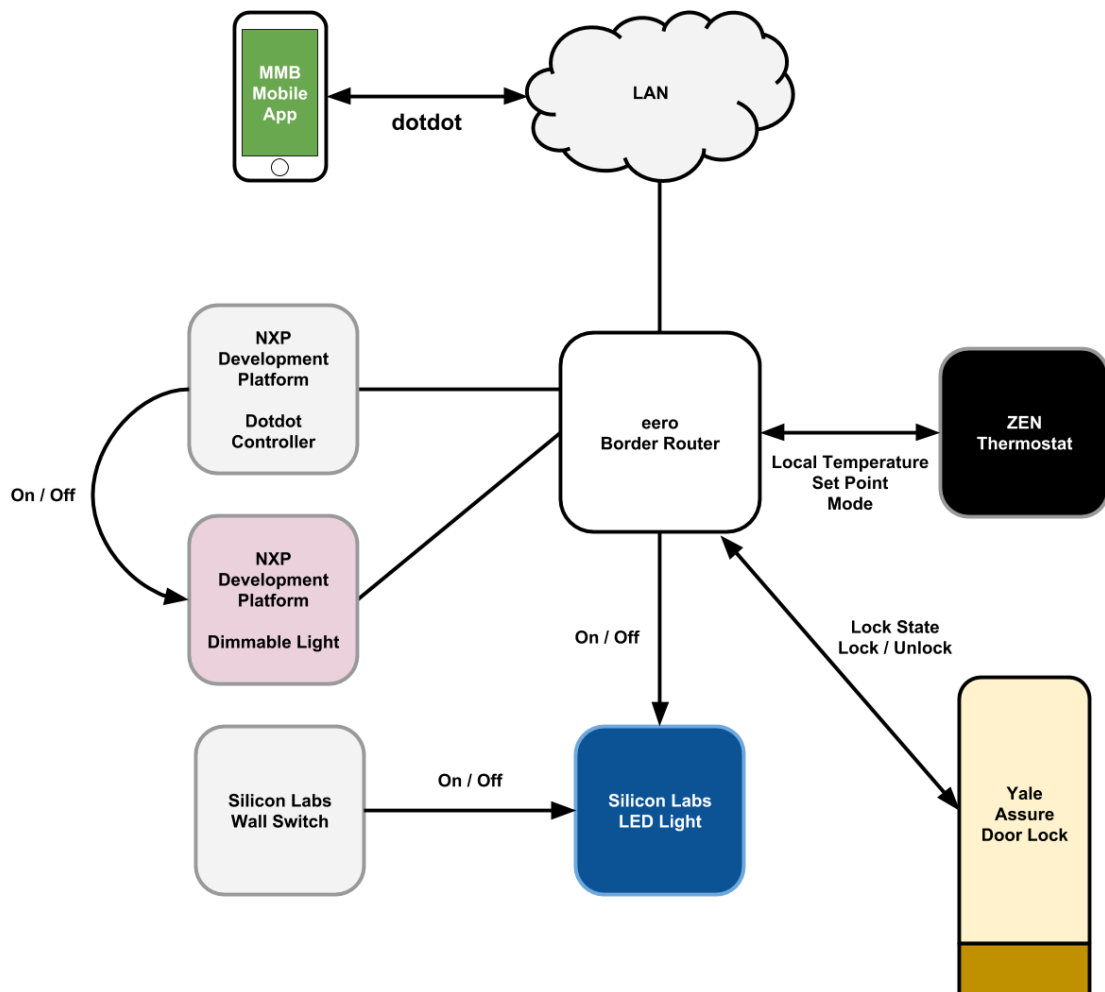


Figure 1 - Dotdot Use Cases

B. Lightweight M2M Use Cases

ARM and NXP demonstrated two Lightweight M2M Use Cases, referred to in the illustrations below as Scenario A and Scenario B.

In Scenario A, the eero border router was joined to the LAN and served as the gateway to the Thread network. Standard Thread commissioning was demonstrated by joining Device 1 (Development Board with NXP K64F + Atmel Radio) and Device 2 (Development Board with NXP K64F + Atmel Radio) via the Thread commissioning mobile application.

The LED on Device 1 was turned on and off using LWM2M On / Off commands.

Scenario A (eero / NXP)

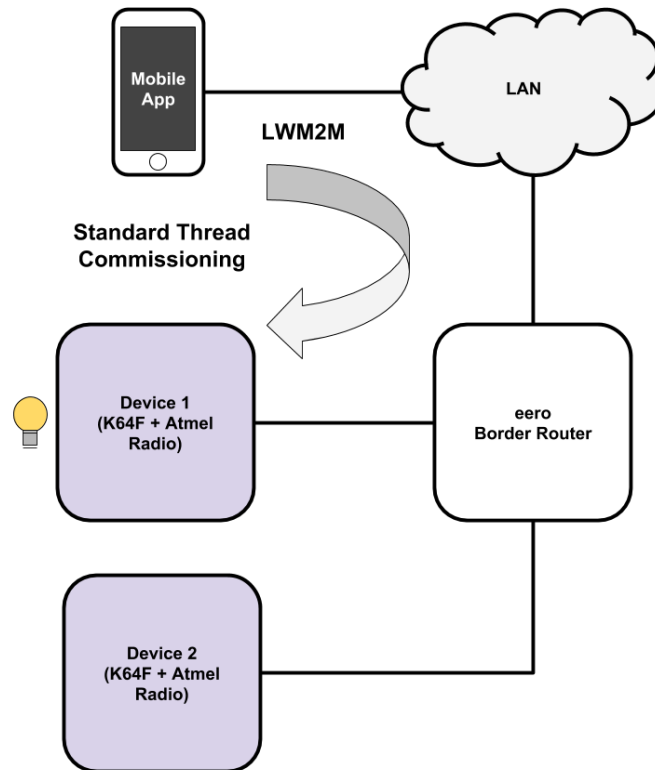


Figure 2 - LWM2M Use Cases, Network A

In Scenario B, an NXP K64F + Atmel Radio development board, configured as a border router, was joined to the LAN and served as the gateway to the Thread network.

Devices 3 through 6, as well as two Thread-enabled LED arrays constructed by Arm, were commissioned on to the Thread network via the border router.

The LED arrays bound groupings of LED's to functional endpoints on the network. LWM2M On / Off commands were issued from multiple points in the network, including the Arm tablet application and via button presses on Devices 3-6, in order to control the LED arrays in various ways. These commands were multi-cast, such that they were applied to target groupings of LED's. This demonstrated the selective application of commands to common device types on a single network.

Scenario B (ARM)

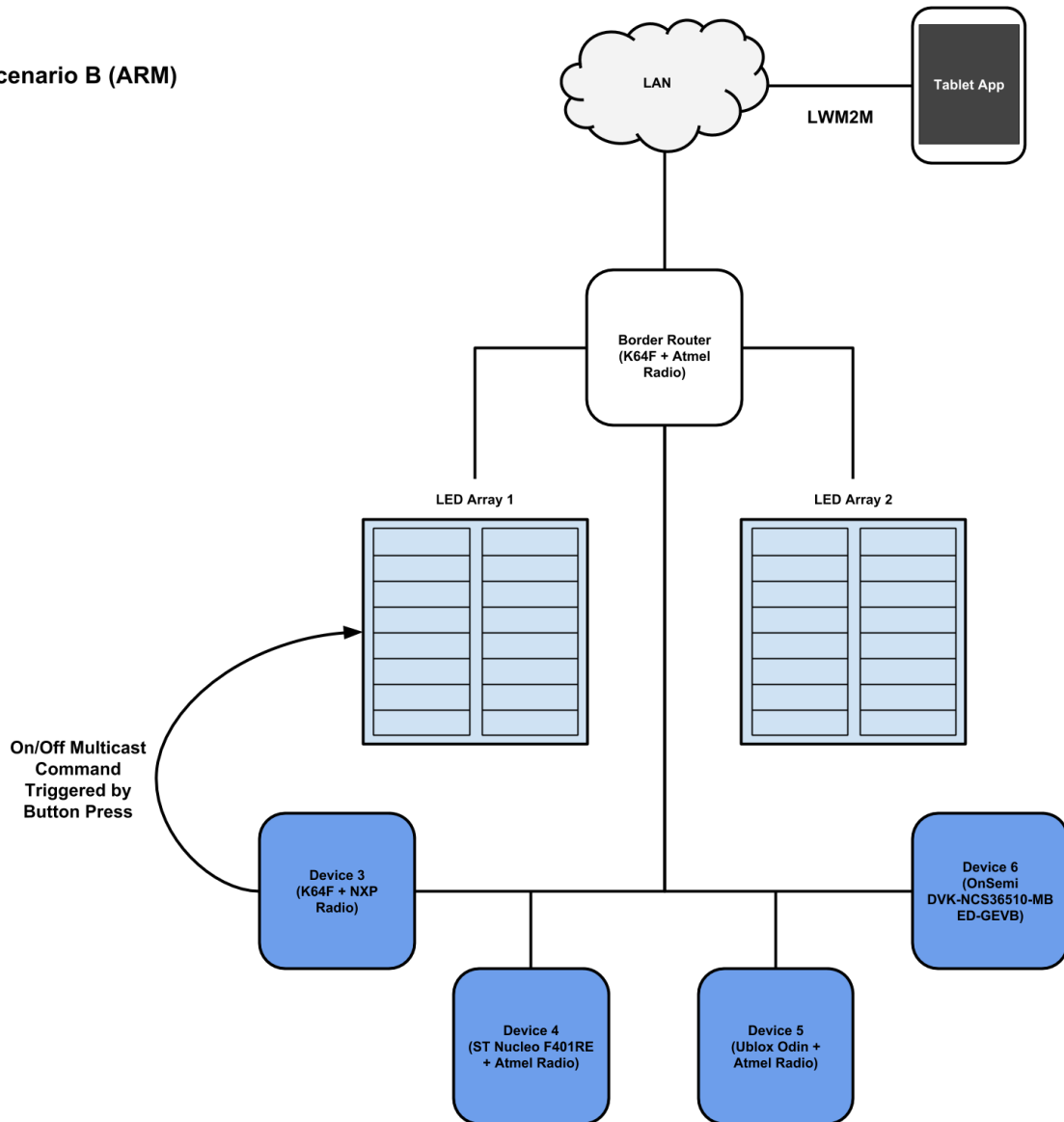


Figure 3 - LWM2M Use Cases, Network B

VII. Conclusion

Through the generous participation of Thread Group members ranging from small private organizations to publicly traded silicon vendors, the Application Workgroup was able to facilitate an effective demonstration of multiple application layers operating on one or more Thread networks.

The interop sought to emphasize the rich suite of open application layer standards able to operate on the Thread network layer. In addition to demonstrating the successful coexistence of these different application layers on the same network, the interop revisited fundamental Thread concepts like device commissioning through mobile applications and the border router. The inclusion of diagnostics tools throughout the interop confirmed that effective, market-ready tools for developers and systems integrators working to implement solutions on Thread today.