Thread Border Router White Paper

July 2022

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.

Thread Border Router White Paper

July 2022

Revision History

Revision	Date	Comments
1.0	July 19, 2022	Public Release

Table of Contents:

1. Authors: Gabe Kassel, Nathan Dyck, Jonathan Hui

2. Introduction

This whitepaper provides an overview of the importance of a Border Router to a Thread Network. We'll walk you through what Thread is, why a Border Router is needed, why Border Routers are important to emerging smart home protocols like Matter, the specific Border Router features that have been standardized in Thread 1.3.0, and how you can get started with your own Thread Border Router.

3. What is Thread

Thread is a low-powered mesh networking protocol based on the IEEE 802.15.4 PHY[1]. It uses 6LoWPAN[2] to ensure messages going into and coming out of the Thread mesh are directly compatible with devices that utilize IPv6 networking; this includes things like personal computers, smartphones, smart speakers, and the broader internet.

To create a Thread mesh, there are a number of roles that devices can play. Devices can dynamically move between certain roles based on the needs of the network:

1. Mesh Extenders: Thread mesh extenders are responsible for routing packets across the Thread mesh.
2. End Devices: End devices always depend on a parent Mesh Extender for receiving messages. There are three categories of end-devices:
    a. Mesh Extenders (REED): Devices that are capable of extending the Thread mesh are known as "Router-eligible end devices" (REED) in the specification. Depending on the needs of the mesh, they may be either Mesh Extenders, or simply End Devices

[1] https://standards.ieee.org/ieee/802.15.4/7029/

[2] https://datatracker.ietf.org/wg/6lowpan/documents/

   b. Minimal end devices (MED): minimal end devices are end devices that cannot become mesh extenders due to design decisions or product constraints. Minimal end devices are always ready to receive messages
   c. Sleepy end devices (SED): sleepy end devices are minimal end devices that are able to sleep in order to preserve power. They will periodically wake to check for messages from their parent. There is a special subclass of sleepy end devices called Synchronized Sleepy End Devices that schedule time with their parent to receive messages. These devices are able to receive messages without transmitting, which can save energy and reduce latency of receiving messages.
3. Border Router: a Border Router is a device that is able to route packets to and from the Thread mesh. For consumers, the most common scenario involves routing between Wi-Fi and Thread, but in principle, a Border Router can route between any IP-bearing interface and the Thread mesh. Despite its name, a Border Router is not required to also be a Thread Mesh Extender, although in practice, it very likely will be.

4. What is a Border Router

A Border Router is a device that is able to route packets to and from the mesh. This routing happens between the Thread mesh and any other IP-bearing interfaces like Wi-Fi, Ethernet, and Cellular.

This routing is very important to ensure Thread devices can communicate with devices that are not natively connected to the Thread mesh. In the consumer space, this might be a Thread light being controlled by a smartphone or smart speaker that doesn't have a Thread radio. In the smart building space, there might be a mainframe computer that needs to communicate with the Thread device to control it or to collect data.

4.1. Network Layer vs. Application Layer

Thread is a successor of Zigbee, a closely-related technology. Zigbee network installations require a network coordinator, which is often packaged into a device that can connect to both Zigbee and other important networks like Wi-Fi, Ethernet, or Cellular. These devices are typically called “hubs”, since all networking communications go through them. One drawback of the Zigbee approach is that the networking layer (how messages are sent) is tied very closely with the application layer (what messages mean). This makes it very hard to standardize a hub in a way that is easily shared between vendors. This means that in the same home or business, you might see multiple, different Zigbee networks that all require their own respective hubs, despite that they *should* be interoperable.

One of the primary benefits of Thread is that it’s based on the Internet Protocol (IP). IP technology is the reason the Internet is so powerful, and Thread extends that to low-power devices. IP technology provides the networking layer for a Thread mesh, which makes standardizing a Border Router possible where it wasn’t before. As a technology, Thread is primarily concerned with the networking layer, which means that any IP-based application layer can be used with a Thread mesh. This means a variety of different apps can share a Thread mesh, in the same way they share Wi-Fi networks and commercial networks today. Any Border Router can be used to route the networking messages for application layers like HomeKit with Thread, Matter with Thread, manufacturer-specific protocols, and future apps still to be invented. Similar to how your internet provider’s modem connects your home network to the internet, a Thread Border Router connects your Thread Mesh to your home network.

Another important difference of Border Routers compared to hubs is the concept of redundancy. A hub, by definition, is a single node through which all communications flow. On the other hand, for a shared Thread Mesh, there can be multiple Border Routers that all provide the same routing capabilities redundantly. If one device fails, the others provide seamless, unbroken connectivity to the mesh.

4.2. Border Router Device Types

Another benefit of Thread Border Routers is that they are simple for companies to build into existing product types—there is no need for yet another new category of products for the smart home. These might be products like smart TVs, smart speakers, smart lighting devices, or Wi-Fi routers.

5. Thread & Matter

Thread provides the networking layer for low-powered IP devices, but that's typically only half the story for building devices that actually provide value to users. The other half is the application layer.

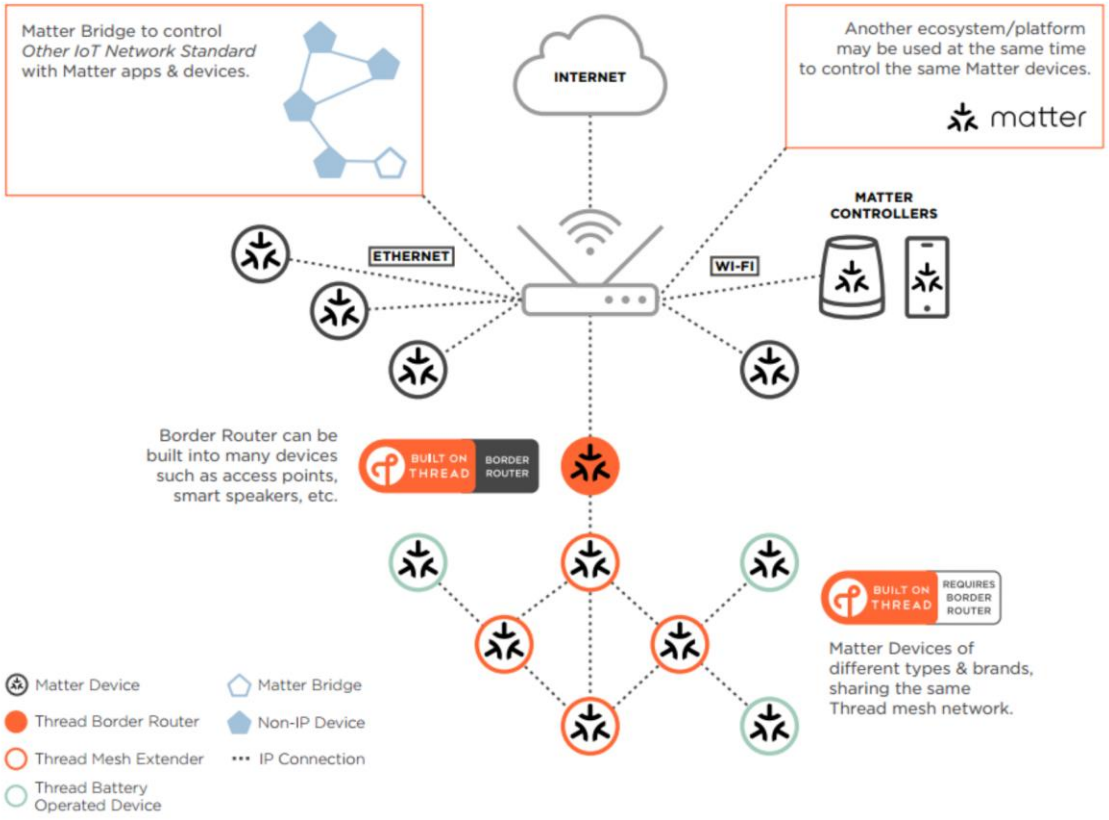
Matter is a very important smart home application layer standard that is based around native IPv6 networking. This brings much-needed interoperability to IPv6-based smart home devices including Thread, Wi-Fi, and Ethernet.

In Matter, devices are paired to a controller known as an Admin. Devices become interoperable when they are paired with the same Admin. Examples of Admins include popular smart home platforms and apps that are interested in controlling, configuring and listening to smart home devices. Groups of such devices are known as a *fabric* of devices, since the IPv6 messages can be woven across many different physical networks like Thread, Wi-Fi, and Ethernet.

This woven nature of Matter Fabrics is why Thread Border Routers are so crucial to the success of Matter with Thread devices, in particular. In principle, a Thread mesh without a Border Router can use Matter if configured properly; however, in practice, a Border Router is ***required*** to enable the majority of use-cases. This is especially true today, since the vast majority of smartphones, smart speakers, and personal computers do not have direct Thread capabilities, so they must communicate with Matter over Thread devices through a Border Router.

One key benefit of Matter is the concept of Multi-Admin. This allows a Matter device to be controlled by separate Admins that may be managed by

different organizations. This concept of Multi-Admin complements the concept of a Shared Mesh enabled by Thread Border Routers. Thread decouples the application layer from the networking layer, meaning that devices that run a Matter-Admin app (e.g. Matter “hubs”) do not *need* to be the same device providing Thread Border Routing. Just the same way your Wi-Fi router doesn’t need to be the device controlling your smart home automation, your Thread Border Router also doesn’t need to be your Matter-Admin. Of course, as a consumer, having things all in one package is nice and convenient; but the key point here is that they are not the same thing: the networking layer is decoupled from the application layer so these devices and roles can be separate.



6. Thread Border Router Technical Features

Thread 1.3.0 standardizes and certifies important features for Border Routers to ensure excellent interoperability and optimal performance. These include:

1. Bidirectional IPv6 Connectivity
2. Service Discovery
3. Transmission Control Protocol (TCP)

6.1. Bidirectional IPv6 Connectivity

A Border Router enables bi-directional IPv6 communication across Thread and non-Thread networks (e.g. Wi-Fi and Ethernet). The Thread, Wi-Fi, or Ethernet networks may or may not already be configured to support IPv6. The Border Router is responsible for detecting the current IPv6 configuration and automatically configuring the networks to enable such communication. This includes IPv6 on-link prefix configuration to configure devices with routable IPv6 addresses and route configuration to configure devices with appropriate routes.

6.2. Service Discovery

A Border Router allows services on the Thread network to be discovered by devices that are not on the Thread network, like your smartphone, smart speaker, or computer. This discovery is accomplished by a technology called mDNS, which is widely used to discover different services (functions) being provided by other devices on a network.

Thread devices that provide a particular “service” or function (e.g. supporting Matter) will register their service(s) with any Thread Border Router on the mesh. Border Routers that are part of the same mesh will respond to discovery queries coming from devices off the mesh and respond on behalf of all the devices on the mesh. This limits the amount of network traffic coming onto the mesh which is important to ensure Thread devices can remain in low-power modes.

6.3. TCP

TCP is a commonly used transport protocol. One of the key features of TCP is congestion control. If the protocol detects that the network is congested, it will slow transmission speed, which is very important to avoid networking bottlenecks at the Border Routers. Support for TCP is primarily the responsibility of the two devices talking to each other (e.g. smartphone and Thread end device), however, certain features of TCP are enabled by ensuring they are supported in certified Thread Border Routers.

7. Sharing Thread Network Credentials

Onboarding a Thread device to a network is generally accomplished by securely giving the network credentials to the device. It is important to understand how this is achieved in order to ensure that Border Routers can actually deliver on the promise of Thread: an extendable low power mesh network with no single point of failure.

First, it's worth pointing out that Thread is still emerging, especially in the consumer space. Sharing network credentials in the Wi-Fi space is well understood by users: there is a human readable SSID and password. In Thread, things are not quite as simple. There is not a single comparable concept for consumer users to latch on to.

The power of Thread as an interoperable protocol is best realized when Border Routers come together and share the same mesh. To accomplish this, there needs to be a path (or multiple paths) for Border Router makers to collaborate and exchange network credentials securely on behalf of a user and ultimately participate in the same Thread mesh.

Access to a Thread network by a device is ultimately gated by the master key, which is a long hexadecimal number that is not human-readable. Because of this, a primary method specified by Thread is the concept of a commissioner passphrase. When coupled with the Thread network name, this is the closest comparable concept to Wi-Fi SSID/password. However, the main difference from Wi-Fi is that the commissioner passphrase only gives indirect access to the mesh. A user with the Thread commissioner passphrase can use this to start an admin session with a Border Router (more generally, Border Agent).

The user (or app) then indirectly tells the Border Router they want to commission a device (e.g. by registering a scanned code). The Border Router then informs the mesh, and any device already on the mesh that is in range of the device to be commissioned can commission it directly over a secure Thread channel.

In the quest for smart home simplicity, this method has not yet been adopted by consumer space deployments including HomeKit and Matter with Thread. Rather than educate the user about the Thread commissioning passphrase, they have instead opted for commissioning Thread devices directly over a secure Bluetooth channel to an already-known network.

This poses a problem for Border Routers. First, if a Border Router is deployed from one manufacturer, how can those credentials be shared to the relevant apps like HomeKit with Thread, Matter with Thread, or other Thread-based application layers? Second, if a Thread mesh already exists, how can a user commission a new Border Router they purchased to the same mesh if they do not know the commissioner passphrase generated automatically for them?

One method to facilitate the sharing of Thread Network credentials is provided by the Mobile OS commissioning APIs, currently available on iOS and on Android. These APIs have been created by their respective mobile OSes in collaboration with the Thread Group; they allow Thread Group members' Apps to add Thread credentials to a user's personal keychain on their mobile phone. Other Thread Group members' apps can read those credentials and commission their own Thread devices in whatever way they choose. General access to these APIs on mobile OSes is open to Thread Group members, although specific access to the APIs by apps always requires user consent.

8. Border Router Tutorial

Getting up and running with your own Thread Border Router is made very straightforward using these instructions from the OpenThread project. OpenThread is an open-source implementation of the Thread networking protocol technology developed and released by Google and maintained on

GitHub. You'll need a Raspberry Pi 4 and an eligible Thread-capable development board.

9. Do I Already Have a Border Router?

As a consumer, having a Border Router is going to be as common as having a Wi-Fi Router in the future. For the now, Border Routers are already in the market, but most of them are not exclusively (or even primarily) a Border Router. Some of them are included in Wi-Fi Routers and Home Gateways; others as part of a smart speaker or display; others as a function of smart lights; and others still, in device types yet to come. So how do you know if you already have one?

Thread Group provides badges to signify a certified device. For end devices, there are two types of badges: one for Border Routers and one for all other Thread devices. Examples are shown below:

You can also check the Thread Certified products list. With the launch of Thread 1.3.0, Border Routers will be certified and listed there.

If you're a little more intrepid and interested in network sleuthing, a tell-tale marker of a Thread Border router is in the mDNS records of your network. Thread Border Routers should advertise a _meshcop._udp. record type, which you can observe using tools like Discovery on iOS or Network Tools on Android. If you see a service with that service type, it means at least one device is acting as a Border Router. Tap on it and you should see information about the device that should include information about it including model number (mn) and manufacturer (vn).

10. Summary

Thread 1.3.0 brings exciting new standardization to the Thread specification that centers around the Border Router. The Border Router, while still

emerging today, is poised to become as ubiquitous as Wi-Fi routers. As a protocol, Thread is uniquely positioned to unleash low-power devices for the smart home and the IoT because it decouples the networking layer from the many application layers that can run on top of the Internet Protocol. This strength shines in the consumer space, allowing certified Border Routers to serve multiple apps like HomeKit with Thread and Matter with Thread. It also unlocks the commercial space with apps like KNX IoT and DALI+. The key components that Thread 1.3.0 standardizes around Border Routers include bi-directional IP connectivity, service discovery, and improved support for TCP in a Thread mesh.

The Thread Group is excited to see more and more Border Routers enter the market and are eager to support you on your journey. If you have additional questions, don't hesitate to contact us at help@threadgroup.org.