HCLTech | Supercharging Progress™

Matter inter-operability for smart connected homes

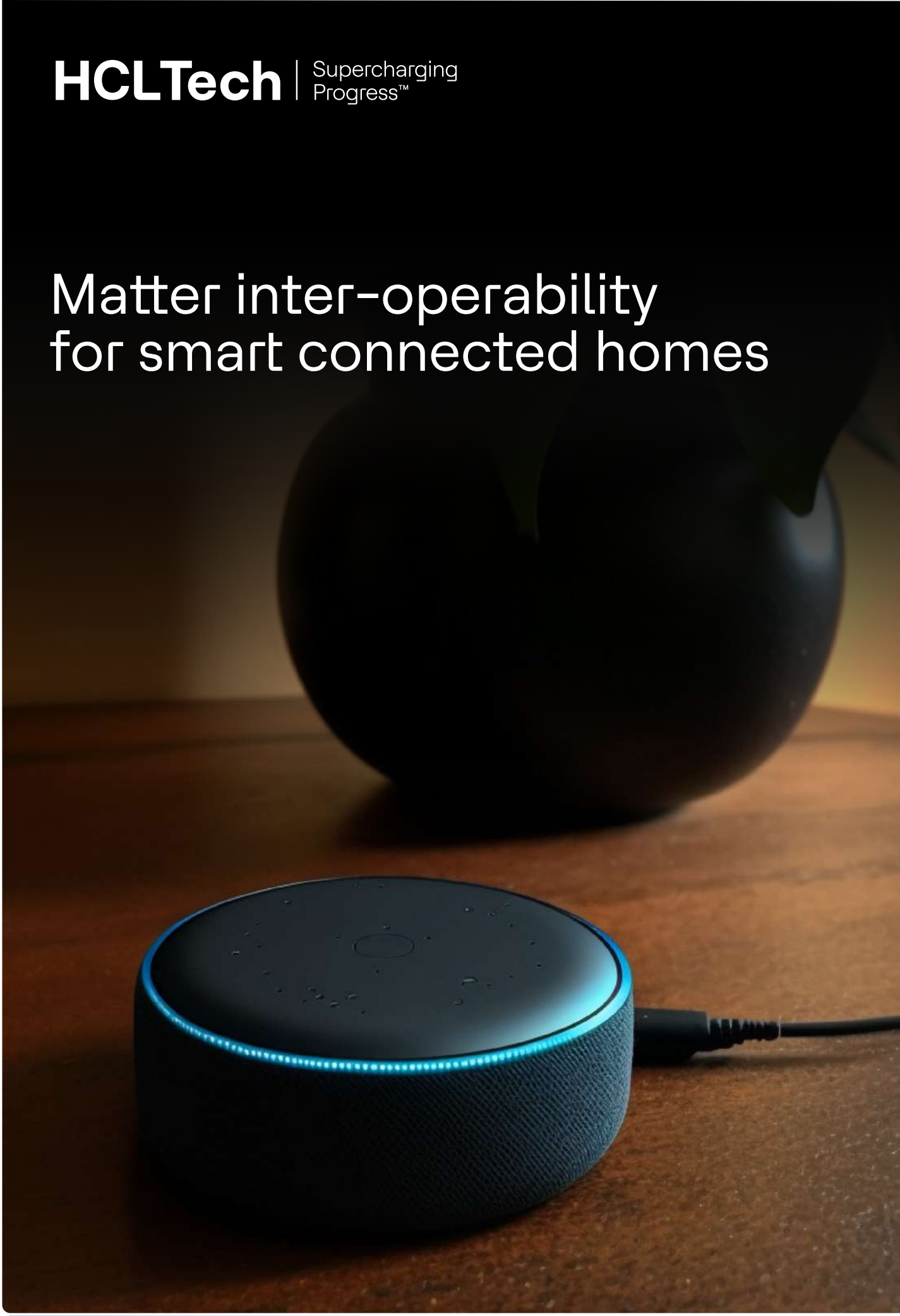

Table of contents

Abbreviations

BLE	Bluetooth Low Energy
ZMQ	Zero Message Queue
SED	Sleepy End Devices
PHY	Physical Layer
PASE	Passcode Authenticated Session Establishment
CASE	Certificate Authenticated Session Establishment
CSR	Certificate Signing Request
NOC	Node Operational Certificate

Introduction

Home automation is a cutting-edge technology empowering homeowners to effortlessly control various aspects of their homes. From lighting and heating to security and entertainment a central control hub facilitates automation by enabling communication between devices through wireless communication protocols like Wi-Fi, Bluetooth, Zigbee, Z-Wave and many more. Voice assistants such as Amazon Alexa and Google Assistant enhance user experience by enabling effortless command execution and real-time retrieval.

The introduction of Matter technology has the potential to revolutionize the home automation industry by establishing a unified, secure and interoperable protocol for smart home devices.

Unlike traditional setups requiring devices from the same manufacturer with compatible technologies, Matter application protocol enables interoperability across different products.

This whitepaper delves into how Matter achieves this inter-operability, including what Matter Fabric is all about, how one can achieve adding a device in the Matter Fabric and what platforms exist to achieve this and a detailed case study of how one can achieve bridging in Matter.

Business challenges

In the realm of smart connected homes, devices like thermostats, a security system, video doorbells, smart locks, security cameras, smart bulbs, smart plugs and smart switches are all integrated and controlled via a mobile application. To address the challenge of interoperability among devices from different manufacturing and utilizing various technologies such as Zigbee devices, Apple's HomeKit devices, Thread devices, Google Weave devices, Z-Wave devices and more, industry leaders collaborated to develop Matter specifications.

The Matter protocol itself is supported on Wi-Fi, Ethernet and Thread technologies. The Matter application protocol defines the application layer that will be deployed on devices as well as link layers to maintain interoperability. The challenges addressed by Matter include:

Interoperability challenge: The lack of a standardized communication protocol poses a challenge for these devices to interact seamlessly. The Matter application protocol addresses the interoperability challenge by encapsulating underlying technologies with the Matter specification.

Need for universal compatibility: The Matter protocol is designed to ensure compatibility across different technologies, including Wi-Fi, Ethernet and Thread enabling devices with varying underlying technologies to communicate effectively within the smart home ecosystem. It establishes a universal IPv6-based communication protocol, providing a standardized framework for smart home devices to communicate seamlessly.

Problem statement

A pressing challenge for smart homes is the requirement for a unified control solution, eliminating the hassle of managing multiple apps, passwords or voice commands. Currently, interoperability issues hinder seamless communication between platforms, such as Alexa unable to interact with Google Assistant or Siri or controlling Google or Apple devices. This raises a critical question: What are the key considerations in developing a singular application capable of seamlessly managing diverse devices from various vendors, each employing distinct underlying technologies?

Matter supported devices

No.	Device	Matter Spec
1	Color temprature light	1.0
2	Contact sensor	1.0
3	Dimmable light	1.0
4	Door lock	1.0
5	Extended color light	1.0
6	Flow sensor	1.0
7	Humidity sensor	1.0
8	Illuminance sensor	1.0
9	Occupancy sensor	1.0
10	On/off light	1.0
11	On/off light switch	1.0
12	On/off plug-in unit	1.0
13	Pressure sensor	1.0
14	Speaker	1.0
15	Temprature sensor	1.0
16	Thermostat	1.0
17	Window Covering	1.0

No.	Device	Matter Spec
1	Refrigerators	1.2
2	Room air conditioners	1.2
3	Dish washers	1.2
4	Laundry washers	1.2
5	Robotic vacuums	1.2
6	Smoke and carbon monoxide alarms	1.2
7	Air quality sensors	1.2
8	Air purifiers	1.2
9	Fans	1.2

Table 1: Supported Matter devices [4] [5]

Matter 1.0 Specification-supported devices include light switches, door locks, window coverings and many more. Matter 1.2 Specification recently added nine new devices to the list.

Matter Fabric

The Matter Fabric employs a Star network topology, where multiple peripheral networks to converge through a central hub network. Within the Matter framework, different types of nodes are defined:

End nodes (Thread Sleepy): Thread Sleepy End Devices (SED) are typically battery-powered devices that connect to another Thread edge node or Border Router. They operate in an idle sleep to conserve power mode and periodically wake up to process any pending data.

Edge nodes (Thread Routing or Wi-Fi): These devices are consistently powered and are connected to either Wi-Fi devices or Thread Mesh Extender devices.

Thread border router: This component facilitates the connection between Wi-Fi and Thread networks and can be integrated into Matter devices. Multiple border routers can be deployed to enhance range and reliability.

Gateway/Hub: This node serves as the bridge between the Matter network and the cloud.

Bridge: This node connects the Matter network to legacy smart home system.

Controller: Responsible for provisioning Matter devices onto the network, ensuring seamless integration and functionality within the ecosystem.

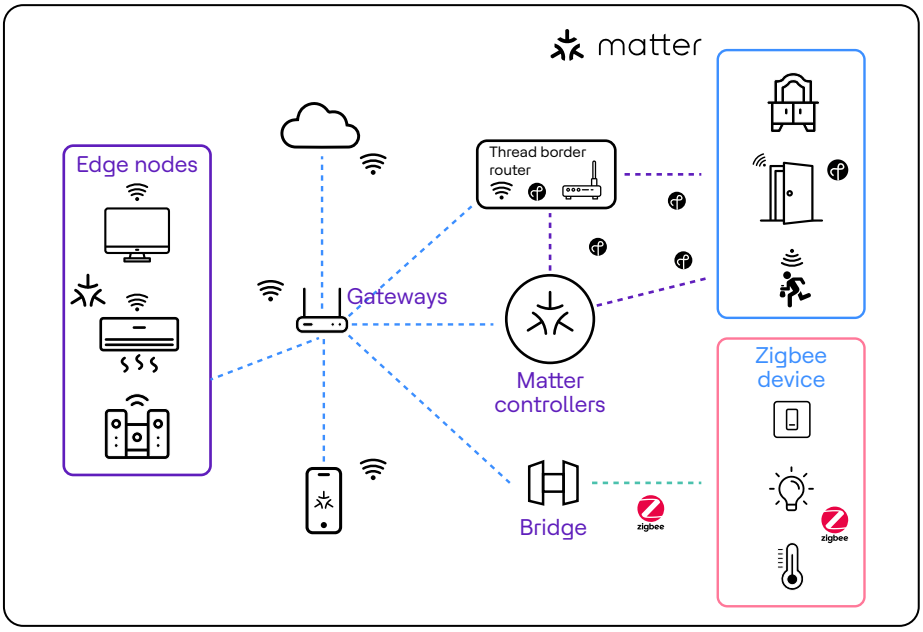

Figure 1: Matter Fabric and Star Topology

The Matter developers can leverage the following (and more) hardware available from different vendors. Typical and similar environment as listed below is used for Matter devices:

Matter devices	Environment	Connectivity	Hardware
Edge nodes	Bare Metal / RTOS	BLE 5.2, Thread, Zigbee 3.0	Cortex M33 1024 KB flash, 296 Kb RAM, BLE, 802.15.4 PHY, Zigbee, Cryptographic acceleration, Debug security, Device identity, Software IP protection
End nodes	Bare Metal / RTOS	Thread, BLE	Cortex M4, 640 KB flash, 152 KB RAM, 128 KB ROM, Bluetooth LE and 2.4 GHz IEEE 802.15.4 compliant, Humidity/temperature sensor, Light sensor/ Gyroscope/6-axis sensor

Thread border router	Bare metal / RTOS	Thread, Wi-Fi	Xtensa dual-core 32-bit LX6 microprocessor, 240 MHz, 448 KB ROM, 520 KB SRAM, 802.11b/g/n, 802.15.4, capacitive touch sensor
Bridge	RTOS /Linux	Thread, Wi-Fi, Zigbee, Legacy home automation link layers	Cortex M0, 256Kb Flash, 32 Kb SRAM, Zigbee, 512MB SDRAM, Wi-Fi, Legacy home automation protocols
Gateway/Hub	Linux	Thread, Wi-Fi, Bluetooth	Cortex A54+Cortex M4, 2GB 32Bit LPDDR4, 16GB eMMC5.1, 32MB Flash, Wi-Fi, Bluetooth
Controller	Linux/Android	Wi-Fi, Thread, Bluetooth	MIPS 24Kc, 650MHz, 128KB D-Cache, 128MB DDR 400MHz, 16MB Flash

Table 2: Supported Matter platforms

The following are the sample architectures for a Matter device, a Matter controller and a Hub device.

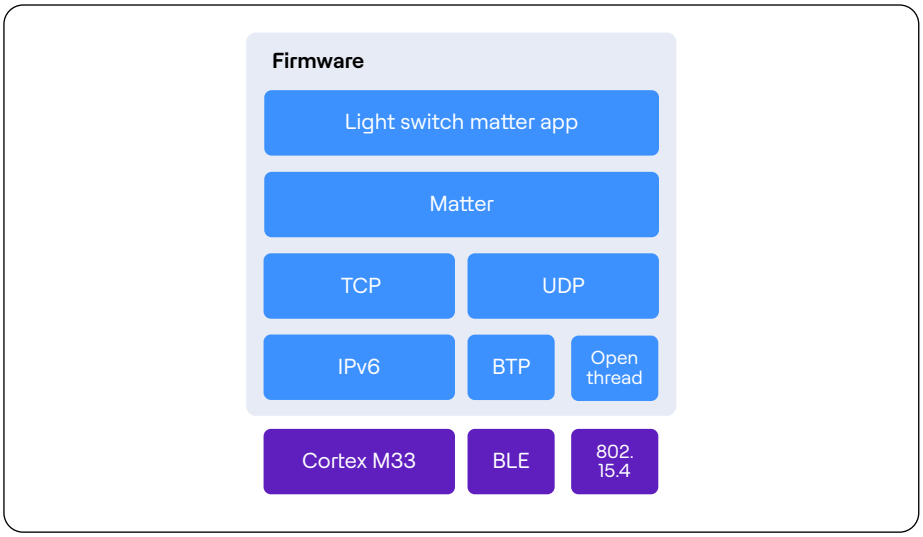

Figure 2: Example architecture for a Bare Metal Matter Edge device

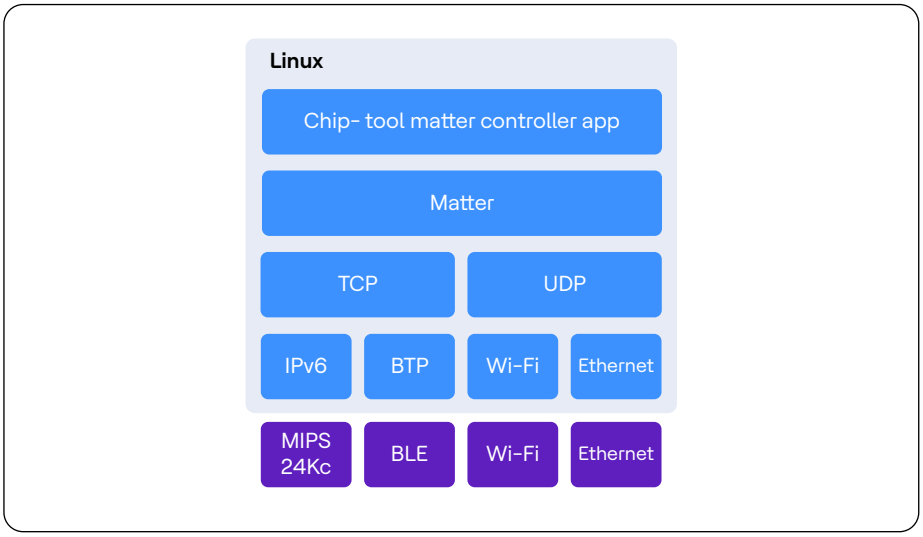

Figure 3: Example architecture for a Linux Matter controller device

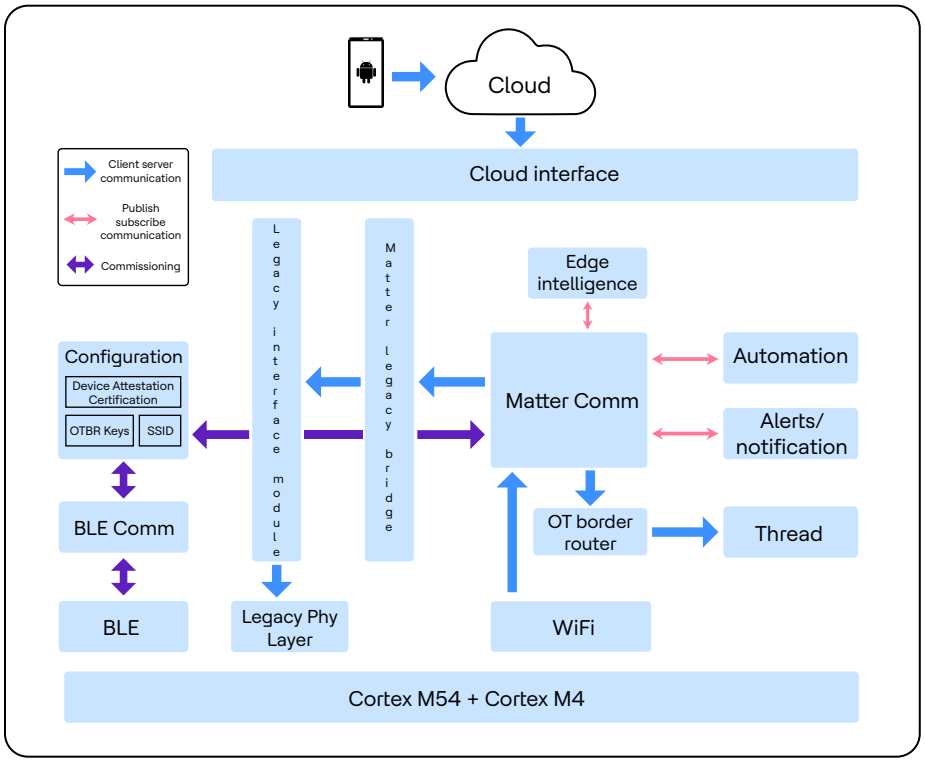

Figure 4: Example architecture for a Linux Matter HUB device

Security through Matter commissioning

Matter devices are assigned a distinct Fabric ID upon joining the Matter Fabric, obtained securely through third-party certification authorities. Their process involves a Matter controller acting as the commissioner, which discovers the device and securely retrieves its information to verify its authenticity. Subsequently, the commissioner installs a Root certificate and assigns a Fabric ID to the commissionee.

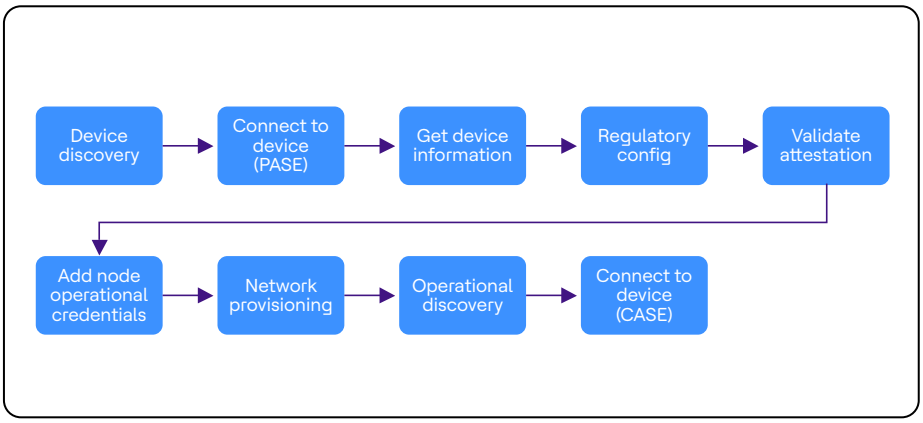

Figure 5: Commissioning in Matter [6]

- **Device discovery:** The commissionee initiates self-advertising through BLE or alternative methods like Wi-Fi/Ethernet (for secondary fabric devices).
- **Connect to device (PASE):** The commissioner establishes a connection to the device by performing Passcode Authenticated Session Establishment (PASE) after matching the discriminator and PIN.
- **Get commissionee information:** The commissioner reads the descriptor cluster, detailing device endpoints and the Basic Information Cluster, providing Vendor ID, Product ID, Product Name and Serial Number after completing the PASE phase.
- **Regulatory configuration:** The commissioner configures the device, addressing parameters like country code and indoor/outdoor location.
- **Commissionee attestation:** The commissioner extracts device attestation and PAI certificates containing Vendor ID, Product ID and attestation public key.
- **Certificate Signing Request (CSR):** The commissioner sends a Certificate Signing Request (CSR) to the commissionee, initiating Certificate Authenticated Session Establishment (CASE) with a unique operational key pair.
- **Add Node Operational Certificate (NOC):** The commissioner installs the Root Certificate on the commissionee.
- **Network provisioning:** Provisioning for the network occurs if the device is not in the primary network.
- **Operational discovery:** Operational discovery facilitates the commissioner and other nodes in the network to identify the commissionee's IP address and port.
- **CASE session establishment:** The exchange of operational certificates occurs, establishing a shared trust by validating they are part of the same logical fabric.
- **Commissioning Complete:** The commissioner utilizes CASE to send a Commissioning Complete command to the newly commissioned device, marking the final step in the commissioning process.

Once successfully completed, the device operates like any other Node on the network.

Solution

In the following sections, we explore the approach to interoperability within smart home devices using the Matter protocol. Examining Figure 6, it becomes apparent that convergence within the cloud and underlying network layers faces obstacles due to:

- Excessive complexity within these layers
- The necessity for constant cloud connectivity
- Incompatible among technologies underneath the model

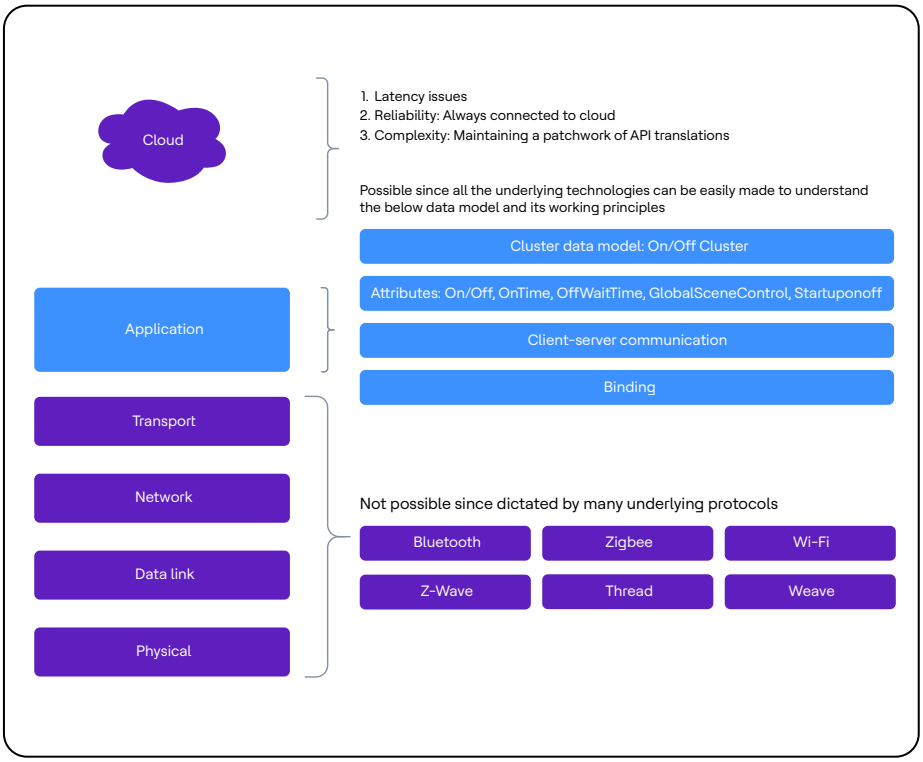

Figure 6: Application layer convergence

In the application layer, the data model and the interaction model of the bridge application on the Matter side align with those on the non-Matter side.

A common approach to achieve convergence involves translating API between the Matter and the non-Matter data model or interaction model. The Matter specification is designed to accommodate various underlying IoT technologies minimizing the burden of conversions on the gateway. For instance, The Zigbee Cluster Library specifies the attributes similar to that of the Matter specification such as the on/off cluster.

To support home automation interoperability, the Matter protocol introduces the concept of bridging devices enabling access to the transport and link layer of other home automation technologies through its data model.

To get into a little more detail in the next section, let us discuss bridging devices for Matter fabric and how the Matter protocol helps in the interoperability with other home automation technologies.

Bridging

The Matter specification provides guidelines for the manufacturers of the bridge device. The manufacturer provides a bridge app that shall interface with both Matter and non-Matter protocols.

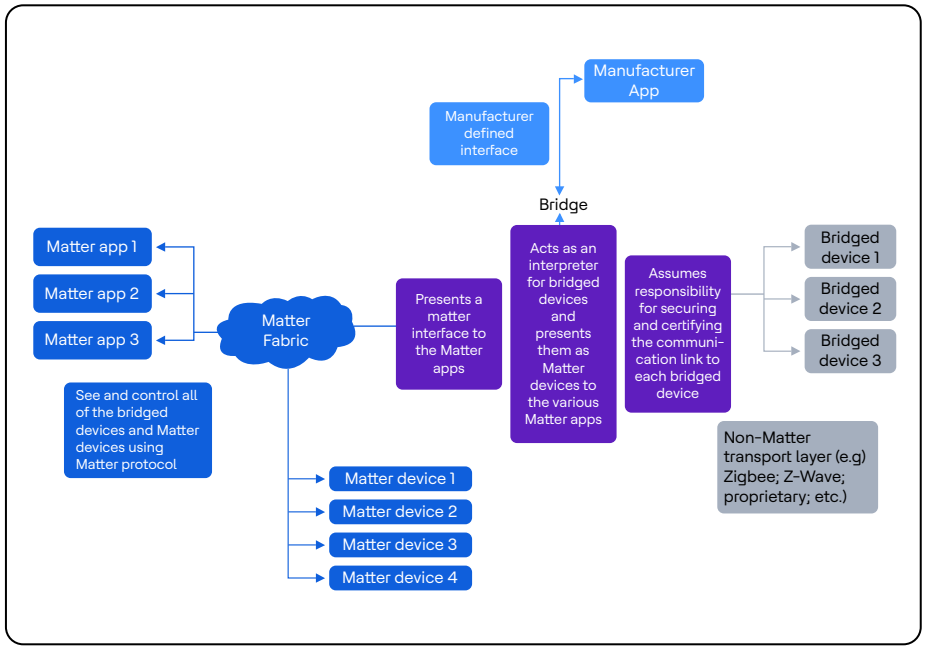

Figure 7: Bridging in Matter [1]

Figure 7 above illustrate that the bridge app primarily holds the responsibilities of commissioning the non-Matter devices into the Matter Fabric, interpreting the data model and interaction model of the non-matter device and mapping them to the Matter technology and securing and certifying the non-Matter communication.

Bridging procedure

This section will explore how the data model concepts can be used by the bridge to expose non-Matter devices in such a way that they can be discovered and used by nodes on the Matter Fabric.

Data model

The data model for the bridge comprises nodes, endpoints, clusters and attributes. It exposes one node to the Matter Fabric, containing various endpoints representing non-Matter devices on the other side of the bridge. Additionally, it exposes an endpoint with the device type Aggregator, which includes a Descriptor cluster with a PartsList attribute containing all the endpoints representing those non-Matter devices. The data model is stored as JSON and tools like zap-tool are available to create the nodes. The image below depicts the data model of the bridge device with Zigbee device endpoints.

Figure 8: Matter data model visualization

Bridge application workflow

Consider a Matter-Zigbee bridge as an example. Figure 9 illustrates that a Matter-Zigbee bridge can be achieved by commissioning the bridge into the Matter network. The required data model for Zigbee devices is modeled as bridged nodes into Matter. Subsequently, the application commissions the bridge device in the Zigbee network as a simple Zigbee router, making it available in both the Matter and Zigbee networks.

The controller node on the Matter network sends commands to the clusters on endpoints of the bridge's node. The bridge receives these commands, comprehends where to forward them after converting between the Matter and the Zigbee protocol. For example, when an On command of Cluster 6 (On/Off Cluster) is triggered from a controller node on the Matter network, the bridge app triggers an On Command on Cluster 6 of the Zigbee device.

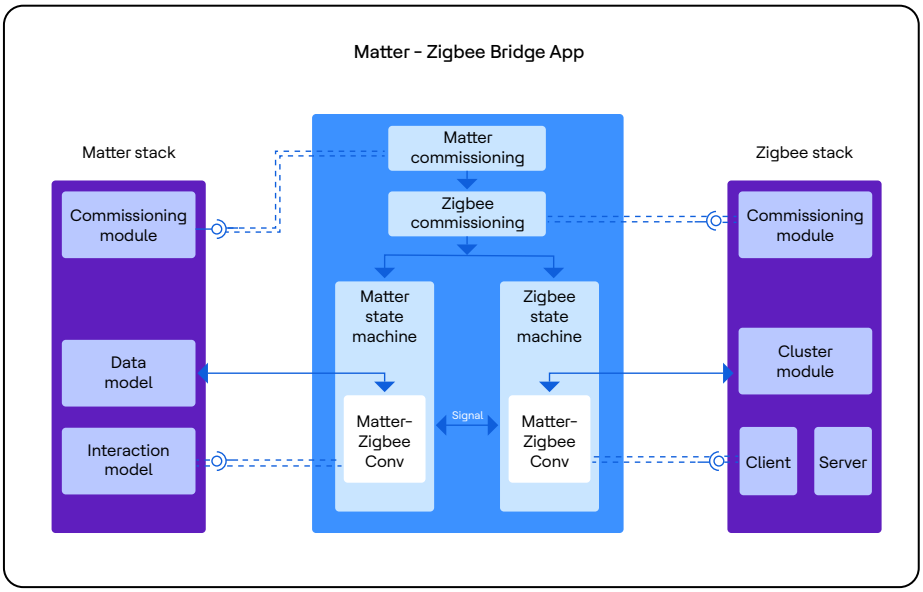

Figure 9: Matter-Zigbee bridging

The bridge application can thus enable Matter interoperability with non-matter protocols. Matter controllers can thus control non-matter devices and hence ease the need to restrict buyers to a particular technology. Now, the underlying technology can be anything; a bridge device resolves the interoperability between the devices.

Implementation challenges and best practices

Implementing a bridge device poses challenges due to the need to be associated with two different technologies, resulting in two separate stacks running on the bridge device. The Matter stack typically follows a Ninja-Style build system, while the non-Matter stack can vary, ranging from static library to open-source code.

One recommended best practice is to employ two distinct applications running on the bridge device interfaced by a ZeroMQ message queue. This approach allows the Matter bridge application with the Matter stack, to signal the non-matter application to change its state or transmit Matter API translation commands.

Alternatively, leveraging the software framework provided by bridge vendors and extending it to suit the required solution is advisable. For instance, the Unify SDK from Silicon Labs offers an open, modular and portable architecture for Matter bridge applications. Matter bridge device support the underlying technologies such as Z-Wave, BLE, Zigbee, Thread and Wi-fi.

Benefits

With the matter interoperability standard, one can envision a smart home with connected devices from different vendors and different underlying technologies and being able to control them all with a single Matter controller or one device can talk to another device of different vendor or technology while preserving their intellectual properties intact.

Matter bridge support for legacy devices

Matter bridges play a crucial role in bridging the gap between modern Matter-enabled smart home devices and legacy technologies like Zigbee and Z-Wave by acting as intermediaries, enabling communication and interoperability between the new Matter standard and existing devices that operate on older protocols. This helps the users to extend the functionality of their smart homes to include legacy devices, fostering a more comprehensive and cohesive ecosystem. The future of these bridges is promising as they facilitate a smooth transition for users with established smart home setups, ensuring compatibility and support for their existing devices.

Third-party device testing for interoperability

Google Smart Home Test Suite offers a comprehensive solution for self-testing and certifying newly added Matter-enabled devices. Initiating the testing process requires a Google Home or a Google Nest hub device for creating a complete smart home test setup that would generate Text-to-Speech (TTS) audio commands corresponding to each device trait reported by action. For accurate validation, these commands must be audible to a nearby Google Home or Nest device, ensuring effective communication with the smart home action. The suite further verifies command results by inspecting the state in the Home Graph to confirm the expected values. Upon completing the test run, results can be submitted through the certification form in the Google Actions console. This comprehensive testing approach ensures the robustness and compatibility of Matter-enabled devices in the Google smart home ecosystem.

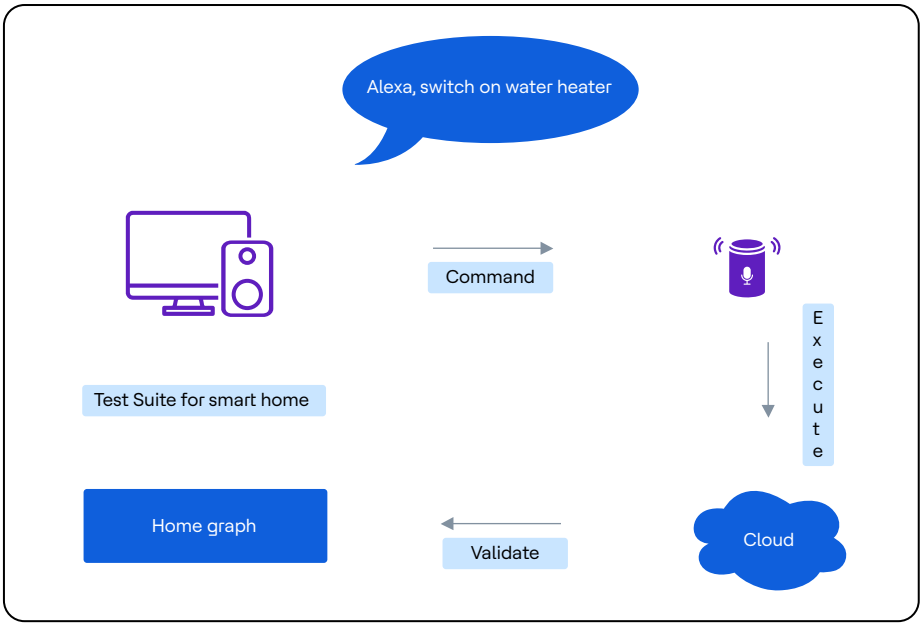

Figure 10: Google Smart Home Test Suite

Conclusion

Matter interoperability standards plays a crucial role in making connected homes smarter by enabling seamless interaction among the devices using the bridging concept.

The Matter specification provides guidelines for the manufacturers of the Bridge device. These manufacturers provide a bridge app that shall interface with both Matter and non-Matter protocols. A typical approach to the implementation of the Bridge device is the API translations between the Matter and the non-Matter data model/interaction model. The bridge application commissions the non-matter devices, representing them in the Matter data model as well as integrates the non-matter protocol. The API translation between the Matter and the non-Matter protocol ensures the interoperability of the devices facilitating smooth communication within the connected home ecosystem.

References

- Matter 1.0 Core Specification
- https://blog.espressif.com/matter-bridge-for-non-matter-devices-d3b7f003a004
- https://www.silabs.com/developers/unify-sdk
- https://developers.home.google.com/matter/supported-devices#:~:text=Table%3A%20Matter%20device%20type%20and%20control%20support%20Legend%3A,%20%20Light%20%2014%20more%20rows%20
- https://csa-iot.org/newsroom/matter-1-2-arrives-with-nine-new-device-types-improvements-across-the-board/#:~:text=The%20new%20device%20types%20supported%20in%20Matter%201.2,fan%20mode%20control%20are%20now%20supported.%20More%20items
- https://developers.home.google.com/matter/primer/commissioning

Author information

Soumya Kolla Srinivasa Murthy

Soumya Kolla Srinivasa Murthy is a Senior Technical Architect with HCLTech. She has extensive experience in developing embedded products for multiple domains. She is a part of the Embedded Platform Lab CoE and has contributed to various projects.

Sathya Durai

Sathya Durai is a Solution Director with HCLTech. He has extensive experience in playing the architect role for embedded software development for products from multiple domains. He is a part of the Embedded Platform Lab CoE and has contributed to various projects.

Koteswaran Srinivasan

Koteswaran Srinivasan is the Technical Director at HCLTech. Koteswaran has more than 20 years of experience in Embedded Product Development and is the Director of the Embedded Platform Lab and Computer Vision and Robotics Lab. He has guided various solution developments in the new areas of Embedded Technologies.