SILICON LABS

Matter Technical Overview

Outline

- **Matter Overview**
- **Key Features**
  - Fabric and Multi-Admin
  - Commissioning
  - Data Model
  - Interaction Model
  - System Model
  - Security
  - Device Attestation
  - DCL
  - OTA Upgrading
- **Q & A**

Matter Overview

Matter Overview (1/8) - Current Landscape of IoT Industry

Consumers

- Extremely hard to mix and match the product they want with their preferred ecosystem
- Very difficult to change once selected

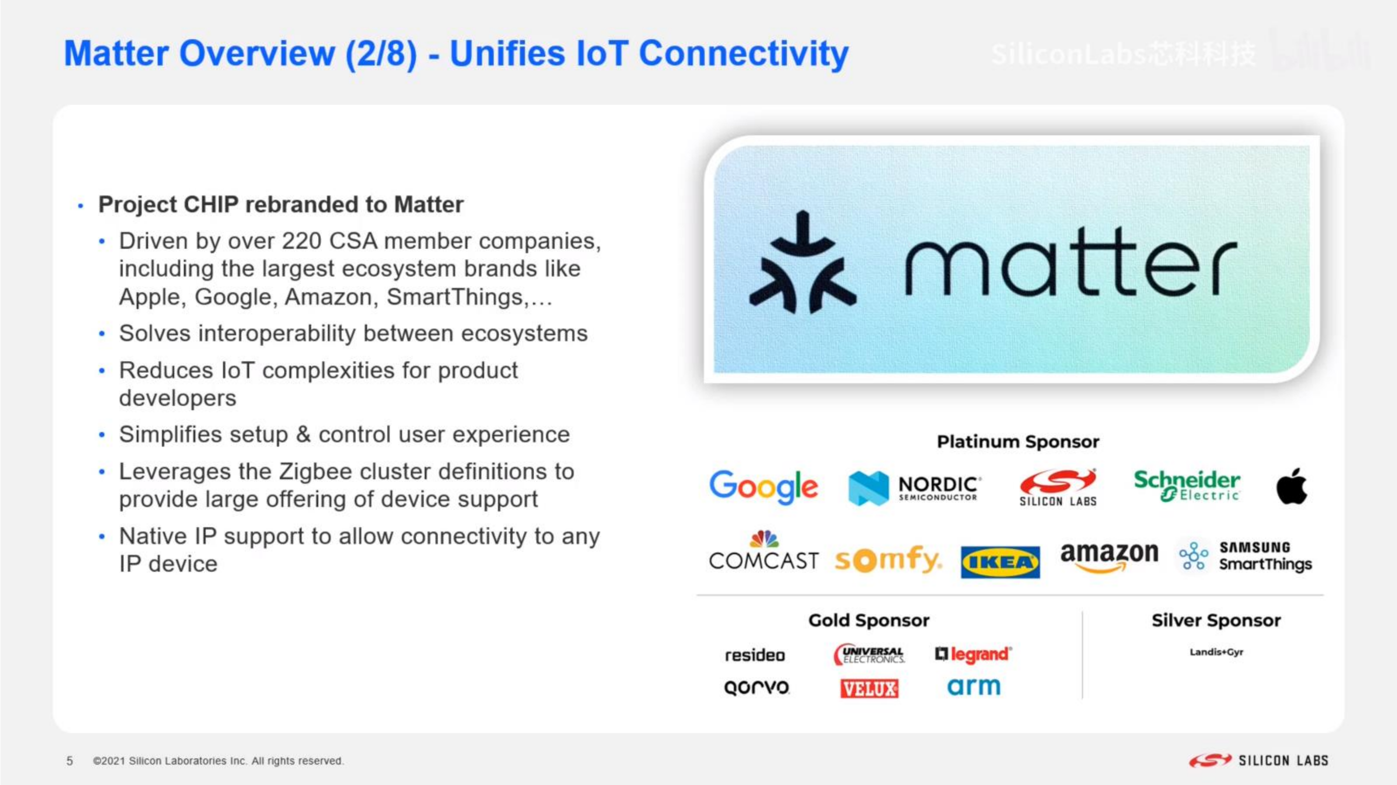
Developers

- Developers are forced to pick what ecosystem integrations they support and often need to ship multiple SKUs for all connectivity standards
- Need to learn different IoT technologies and ecosystems

Retailers

- Too difficult to provide expert advice to answer consumer questions
- High return rates due to interoperability issues

WORKS WITH
amazon alexa
WORKS WITH ring
works with
XFINITY Home
Works with
SmartThings™
works with the
Google Assistant
Friends of
hue
personal wireless lighting
Works with
Apple HomeKit



Matter Overview (2/8) - Unifies IoT Connectivity

- **Project CHIP rebranded to Matter**
  - Driven by over 220 CSA member companies, including the largest ecosystem brands like Apple, Google, Amazon, SmartThings,…
  - Solves interoperability between ecosystems
  - Reduces IoT complexities for product developers
  - Simplifies setup & control user experience
  - Leverages the Zigbee cluster definitions to provide large offering of device support
  - Native IP support to allow connectivity to any IP device

matter

Platinum Sponsor

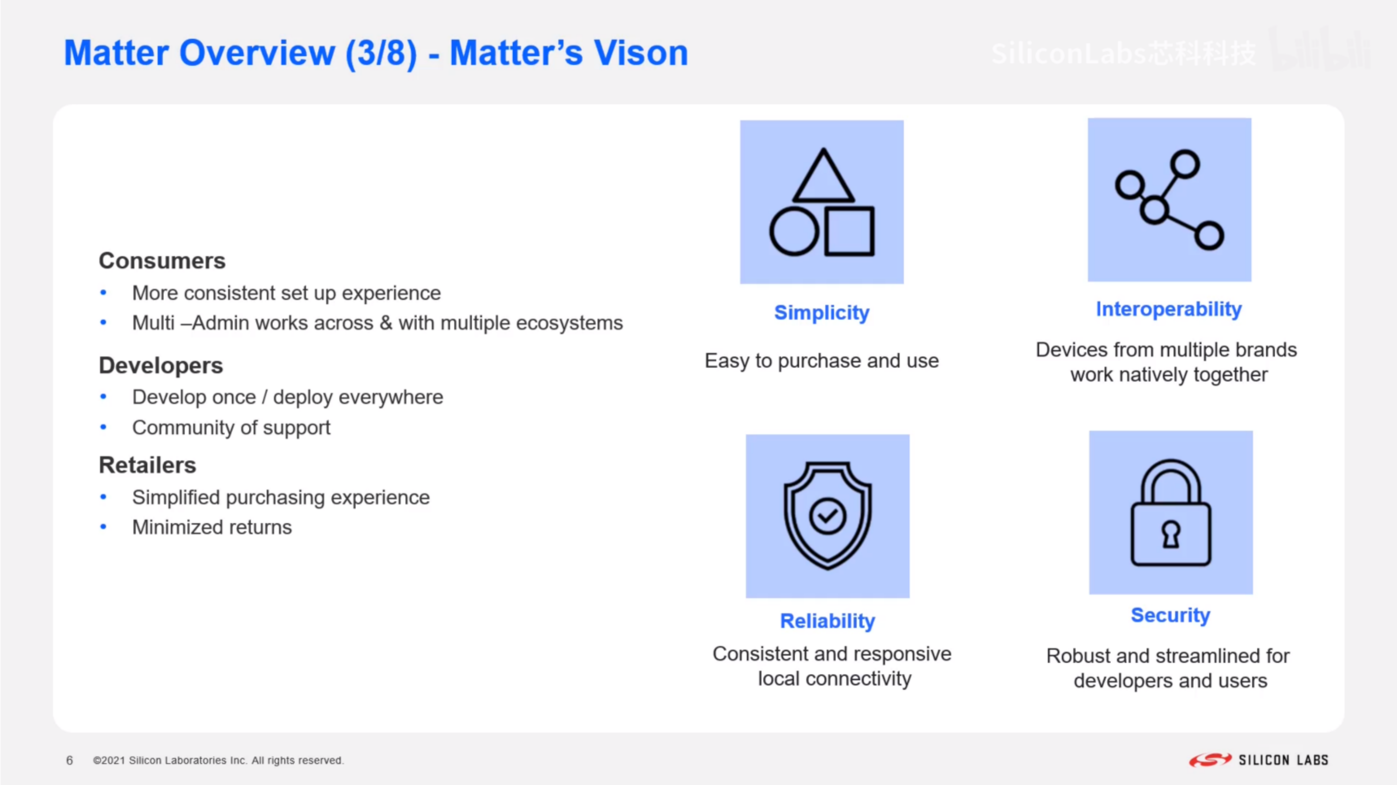
Google, Nordic Semiconductor, Silicon Labs, Schneider Electric, Apple, Comcast, Somfy, IKEA, amazon, Samsung SmartThings

Gold Sponsor

resideo, Universal Electronics, legrand, Qorvo, VELUX, arm

Silver Sponsor

Landis+Gyr



Matter Overview (3/8) - Matter's Vison

Consumers

- More consistent set up experience
- Multi –Admin works across & with multiple ecosystems

Developers

- Develop once / deploy everywhere
- Community of support

Retailers

- Simplified purchasing experience
- Minimized returns

Simplicity

Easy to purchase and use

Interoperability

Devices from multiple brands work natively together

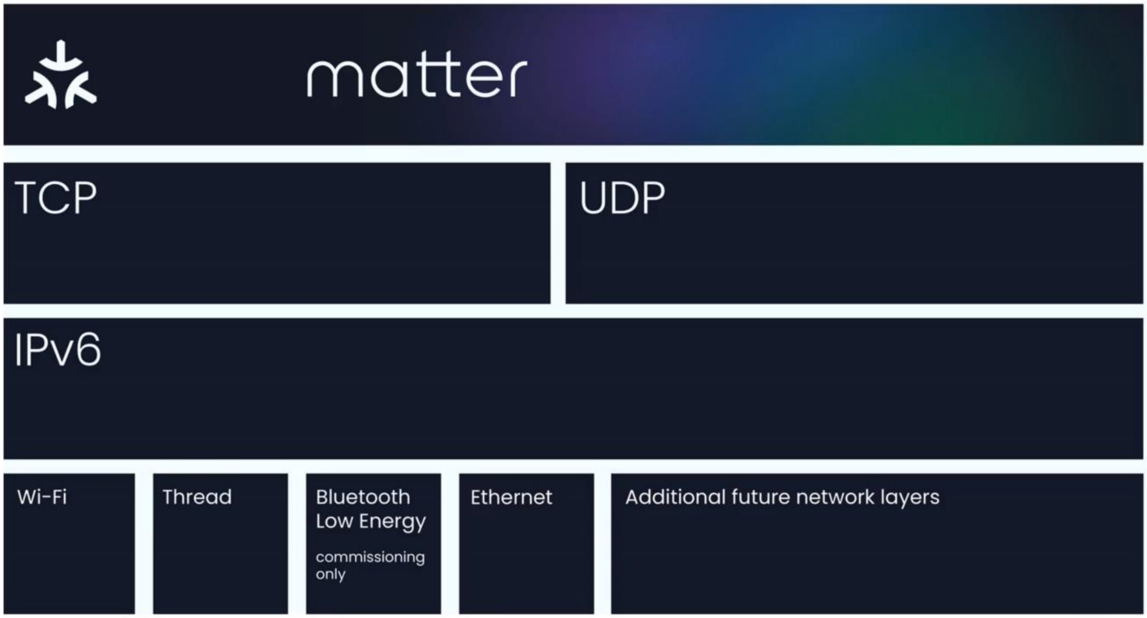
Reliability

Consistent and responsive local connectivity

Security

Robust and streamlined for developers and users

Matter Overview (4/8) - Architecture



matter

TCP

UDP

IPv6

Wi-Fi

Thread

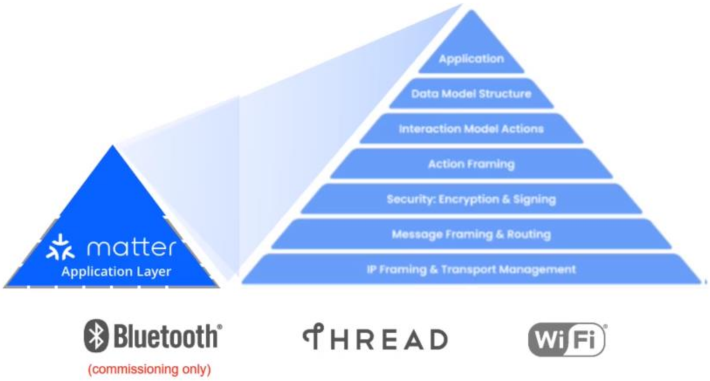
Bluetooth Low Energy

commissioning only

Ethernet

Additional future network layers

Matter Overview (5/8) - Layered Architecture



matter

Application Layer

Application

Data Model Structure

Interaction Model Actions

Action Framing

Security: Encryption & Signing

Message Framing & Routing

IP Framing & Transport Management

Bluetooth

(commissioning only)

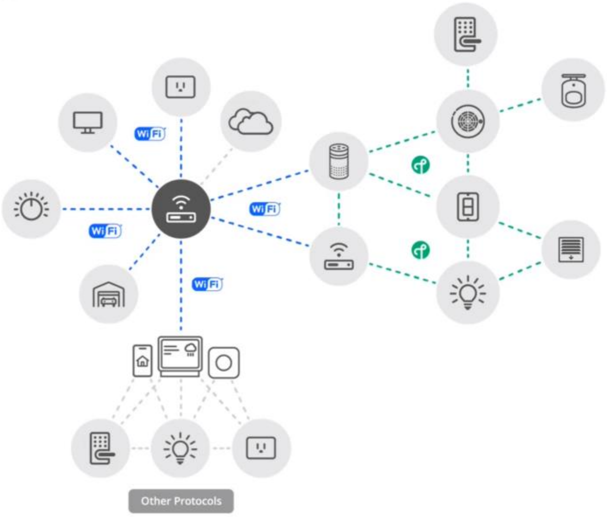
THREAD

WiFi

- **Common application layer + data model**
  - Interoperability, simplified setup & control
- **IP-based**
  - Convergence layer across all compatible networks
- **Secure**
  - AES-128-CCM encryption with 128-bit AES-CBC
- **Open-source development approach**
  - Based on market-proven technologies
- **Common protocol across device and mobile**
  - Extendible to cloud
- **Low overhead**
  - MCU-class compute, <128KB RAM, <1MB Flash

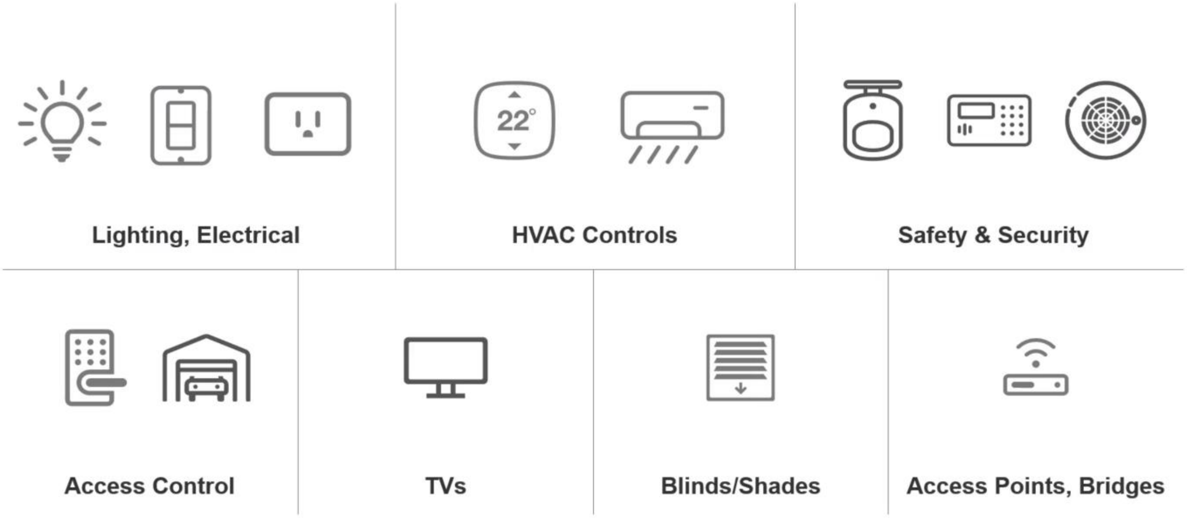
Matter Overview (6/8) - Network Topology

matter



- Focus on Ethernet / WiFi / Thread
- BLE is used as the commissioning channel
- Thread devices connect to other IP networks through border routers
- Bridges can link to other protocols like Zigbee and Z-Wave

Matter Overview (7/8) - Matter Target Applications



Lighting, Electrical

HVAC Controls

Safety & Security

Access Control

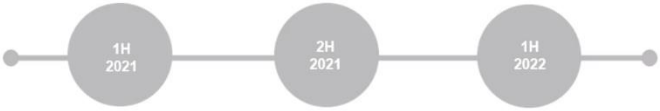
TVs

Blinds/Shades

Access Points, Bridges

Scoping exercises for additional device types and use cases underway and continual.

Matter Overview (8/8) - Schedule & Certification



Initial technical specifications available to Members
Initial SDK and Test Event Efforts

Pre-balloting technical specifications available to members
Ongoing SDK & Cert Program Dev
Test Events Continue

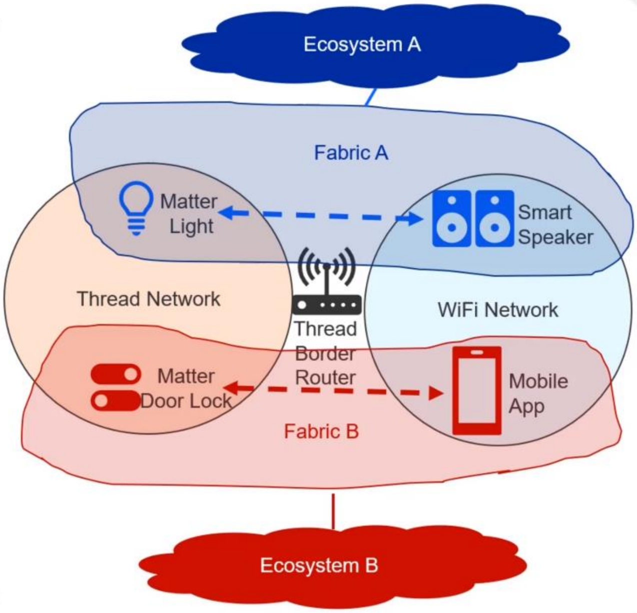
SDK Released
1st Products Certified
Certification Program Released

- Matter logo is a seal of approval that devices will work seamlessly together today & tomorrow
- **Only** certified products can use the Matter name or logo. Test event participants who successfully completed the test events will be the first batch of certified products.
- **Only** members of the CSA will be able to certify their products once the Certification Program is released

Key Features

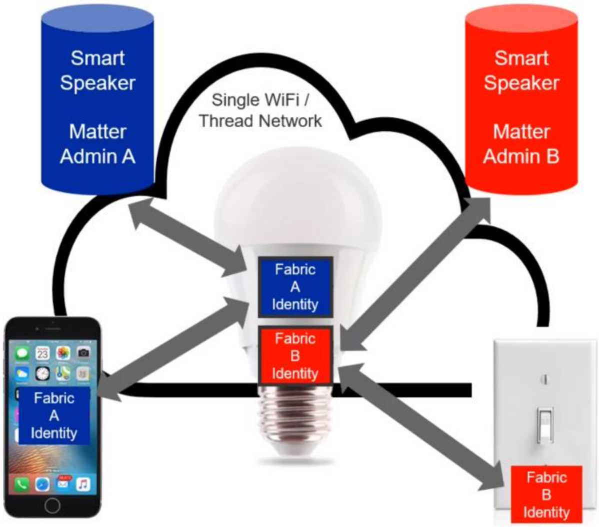
Fabric and Multi-Admin (1/2) - Fabric

- **Fabric**
  - A collection of Matter devices sharing a trusted root
  - A fabric is identified by a **fabric ID** which is a **64-bit number**
- **Node**
  - In a Matter fabric, each physical device is called a node
  - Each node is identified by a **node ID** which is a **64-bit number**



Fabric and Multi-Admin (2/2) - Multi-Admin

- Provides a means for multiple Matter Fabrics and their administrators to manage devices
- Each Matter Fabric can have unique root authority
- Devices must support multiple Matter Admins
- Matter admins dictate the access control lists for their Matter fabric, and thus the devices can access the device
- Example:
  - Matter Admin A can grant control privileges to Smart Phone on Fabric A
  - Matter Admin B can grant control privileges to Smart Switch on Fabric B
- Access Control is **separate** for both fabrics



Commissioning (1/5) - Overview

Supports two potential starting points

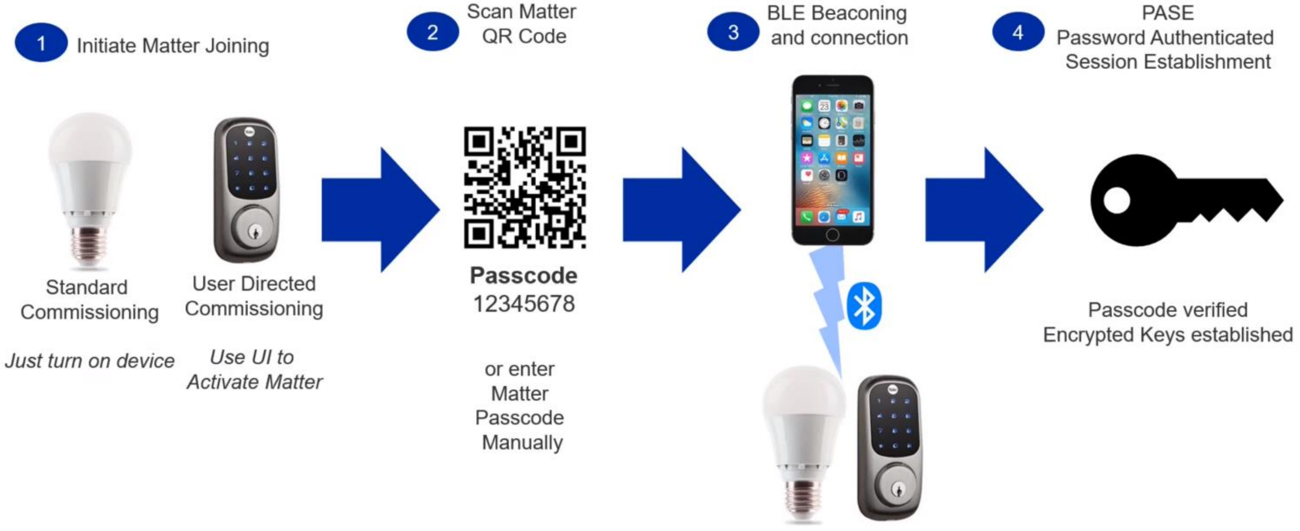
A. **Device already on the network**

B. **Device needs network credentials for WiFi or Thread (Requires BLE support)**

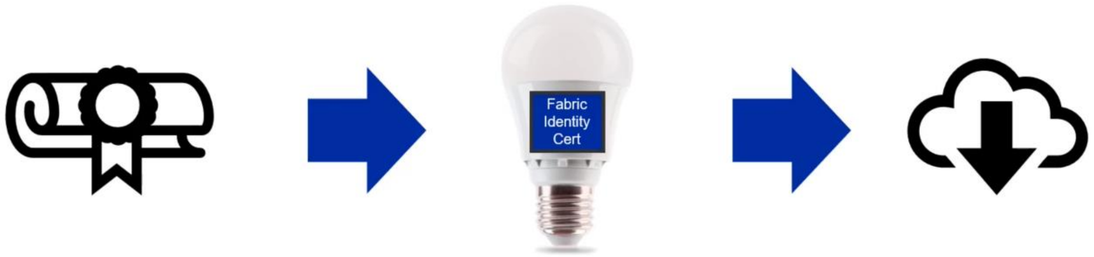
Handles these main commissioning flows

Commissioning Flow Name	**Description**
Standard	Device automatically goes into the commissioning mode on power-up. Beneficial for limited UI devices (e.g. Bulbs)
User Directed	Device only enters commissioning mode as initiated by the user. Helpful for devices that have user interfaces or that want to protect the commissioning mode from being initiated without user present.

Commissioning (2/5) - Sample Commissioning Flow (Part 1)


1 Initiate Matter Joining
2 Scan Matter QR Code
3 BLE Beaconing and connection
4 PASE Password Authenticated Session Establishment
Standard Commissioning
Just turn on device
User Directed Commissioning
Use UI to Activate Matter
Passcode
12345678
or enter Matter Passcode Manually
Passcode verified
Encrypted Keys established

Commissioning (3/5) - Sample Commissioning Flow (Part 2)



5 Device Attestation

Check manufacturer certificate and device compliance

6 Install Operational Security

Install a commissioner root certificate, an operational certificate for device, and an ACL with list of administrators.

7 Configure Operational Network

Convey WiFi or Thread network credentials using Network Commissioning Cluster

Commissioning (4/5) - Sample Commissioning Flow (Part 3)

8 Join Network

Device attaches to WiFi or Thread network

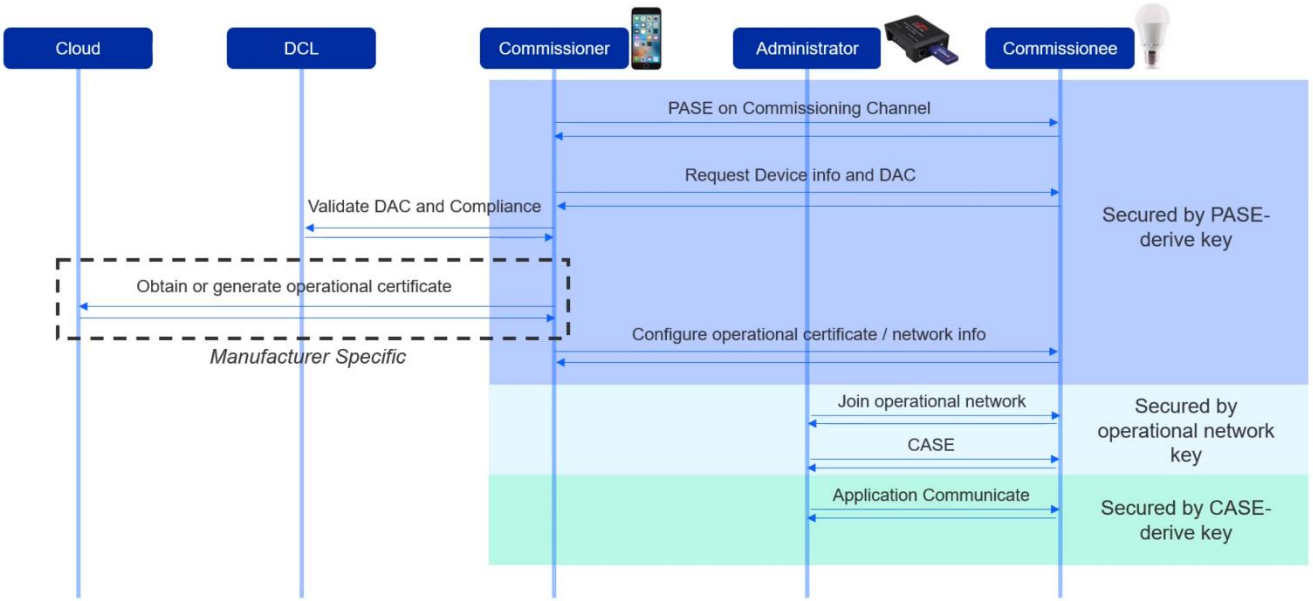
9 Discover Device on IP network

Controller discovers device using DNS and establishes a CASE session using newly configured security material.

10 Finalize

Inform Device Commissioning is complete
Performs any other application configuration

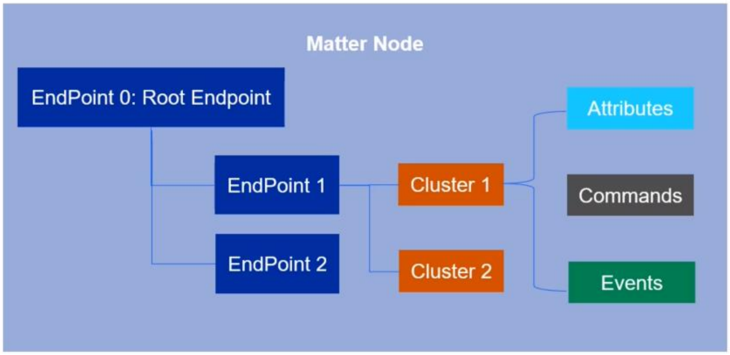
Commissioning (5/5) - Commissioning Flow Overall


Cloud
DCL
Commissioner
Administrator
Commissionee
PASE on Commissioning Channel
Request Device info and DAC
Validate DAC and Compliance
Secured by PASE-derive key
Obtain or generate operational certificate
Configure operational certificate / network info
Manufacturer Specific
Join operational network
CASE
Secured by operational network key
Application Communicate
Secured by CASE-derive key

Data Model (1/4) - Overview

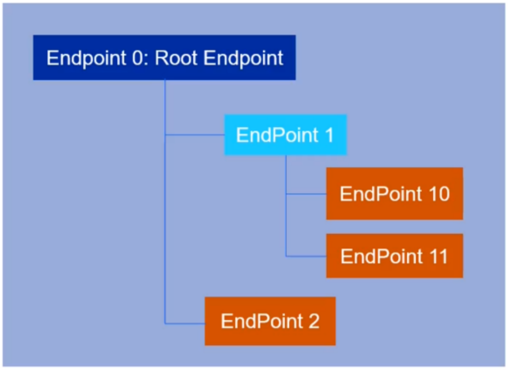
- **Leverages the dotdot Data Model**
  - Logic function unit is represented by endpoint
  - Specific functions are described by clusters
  - Interactions happen between local endpoints and remote endpoints in a client/server model

Note: Does ***not*** perfectly match Zigbee and was extended for new functionality needed by Matter



Data Model (2/4) - Endpoints

- Endpoints are logical device types that can be accessed within the same physical device
- Endpoint 0 is reserved as the **root endpoint**
- Endpoint 0 is **mandatory** for every device
- **Composed endpoint** could be used to implement composed device



Endpoint 0: Root Endpoint

EndPoint 1

EndPoint 10

EndPoint 11

EndPoint 2

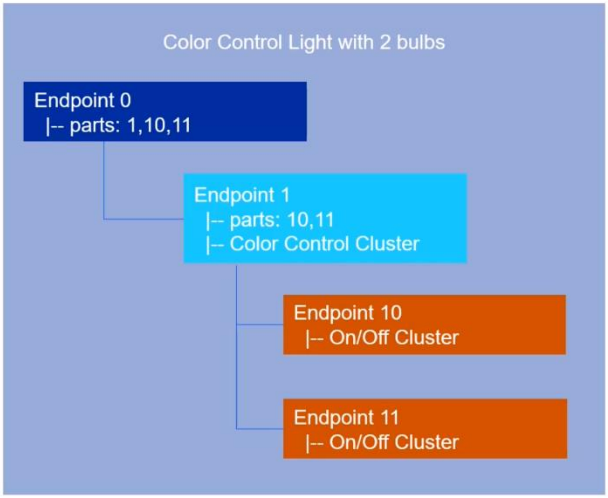
🟦 root node endpoint

🟦 composed endpoint

🟧 leaf endpoint

Data Model (3/4) - Composed Endpoint Example

- **Assuming the requirement is**
  - To develop a Color Control Light with 2 bulbs
    - Each bulb can be turned on/off independently
    - The color of the bulbs must be controlled together



Color Control Light with 2 bulbs

Endpoint 0
|-- parts: 1,10,11

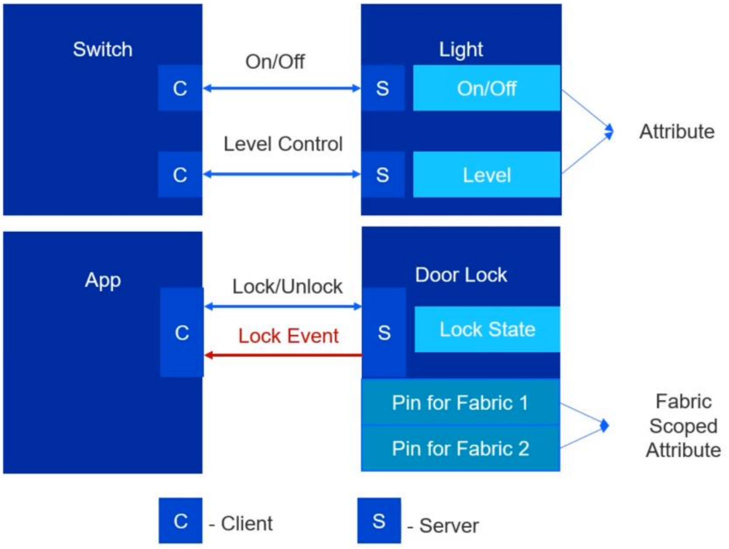
Endpoint 1
|-- parts: 10,11
|-- Color Control Cluster

Endpoint 10
|-- On/Off Cluster

Endpoint 11
|-- On/Off Cluster

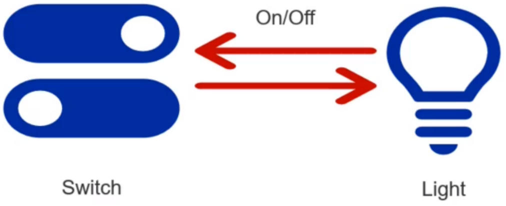
Data Model (4/4) - Cluster

- **Client/Server communication model**
  - Attributes
  - Commands
  - Events
- **Inherited from ZCL (Zigbee Cluster Library)**
- **Security related attributes are fabric scoped**

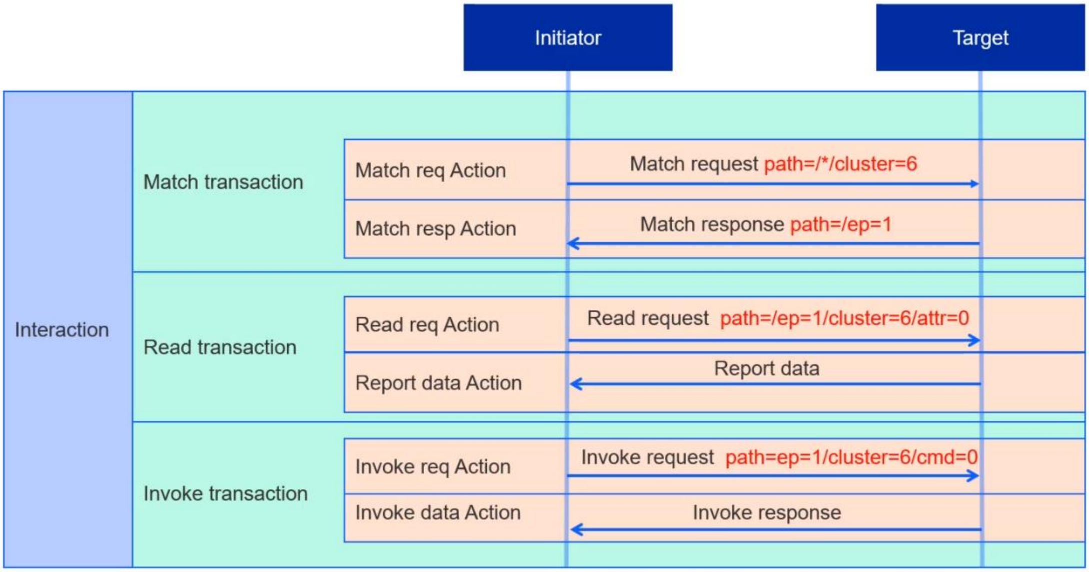


Interaction Model (1/2) - Overview

- **Path**, identify the data to operate
  - Attribute -- <endpoint><cluster><attribute>
  - Command -- <endpoint><cluster><command>
  - Event -- <endpoint><cluster><event>
- **Action**, a request OR a response from the initiator to the target node
  - Read request
  - Report data
  - Subscribe request / response
  - Write request / response
  - Invoke request / response
  - Match request / response
  - Timed request
  - Status response
- **Transaction**, a sequence of one or more actions
- **Interaction**, a sequence of one or more transactions



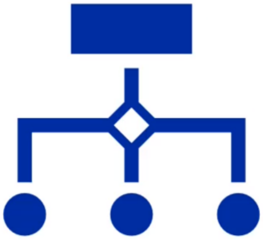
Interaction Model (2/2) - Example



Interaction	Match transaction	Match req Action	Match request path=/*/cluster=6 →
		Match resp Action	← Match response path=/ep=1
	Read transaction	Read req Action	Read request path=/ep=1/cluster=6/attr=0 →
		Report data Action	← Report data
	Invoke transaction	Invoke req Action	Invoke request path=ep=1/cluster=6/cmd=0 →
		Invoke data Action	← Invoke response

System Model (1/8) - Overview

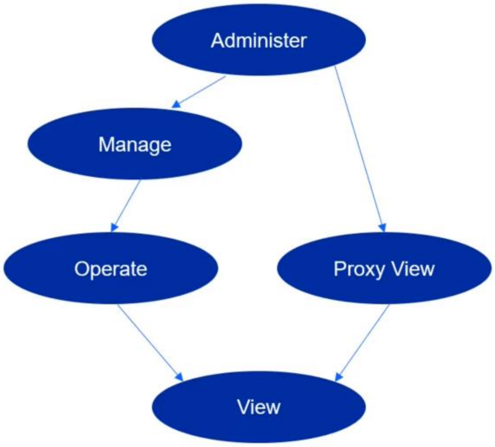
- **System Model**
  - Access Control Cluster
  - Descriptor Cluster
  - Label Cluster
  - Binding Cluster
  - Proxy
  - Bridge



Management Model

System Model (2/8) - Access Control Cluster

- **Describe the access control list of the Node**
- **Only present on endpoint 0**
- **User can control the privilege of accessing the endpoints, clusters of the current node**
  - Administer
    - Manage privileges, can read/observe/modify ACL
  - Manage
    - Can modify configurations except ACL
  - Operate
    - Can view and perform primary functions except ACL
  - Proxy View
    - Can read and observe all
  - View
    - Can read and observe except ACL

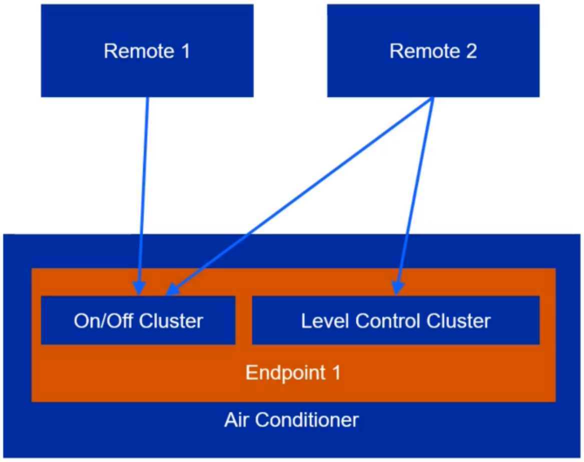


Access Control Privilege Levels

System Model (3/8) - Access Control Cluster Example

- **Assuming the requirement is**
  - Remote 1 can only turn on/off the air conditioner, but can't adjust the temperature
  - Remote 2 can turn on/off the air conditioner as well as adjust the temperature

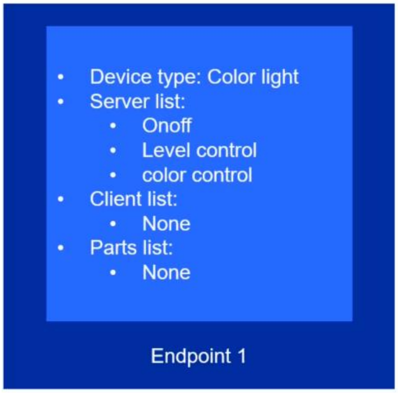
```
ACL: [
  0: {
    FabricIndex: 0,
    Privilege: operate,
    AuthMode: CASE,
    Subjects: [node ID of phone 1],
    Targets: [
      endpoint: 1,
      cluster: "on/off"
    ]
  },
  1: {
    FabricIndex: 0,
    Privilege: operate,
    AuthMode: CASE,
    Subjects: [node ID of phone 2],
    Targets: [
      endpoint: 1,
      cluster: [
        "on/off",
        "level control"
    ]
  }
],
```



Access Control Cluster Example

System Model (4/8) - Descriptor Cluster

- **Attributes**
  - Device type list
  - Server list
  - Client list
  - Parts list



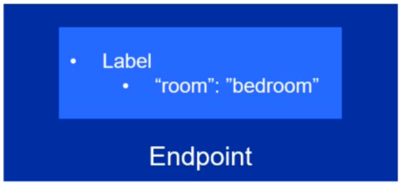
- Device type: Color light
- Server list:
  - Onoff
  - Level control
  - color control
- Client list:
  - None
- Parts list:
  - None

Endpoint 1

System Model (5/8) - Label Cluster

Provides a mechanism to tag the endpoints

- **Attribute**
  - Label list
- **Label list**
  - Label
  - Value
- **Derived Clusters**
  - Fixed label cluster– Read-only label
  - User label cluster

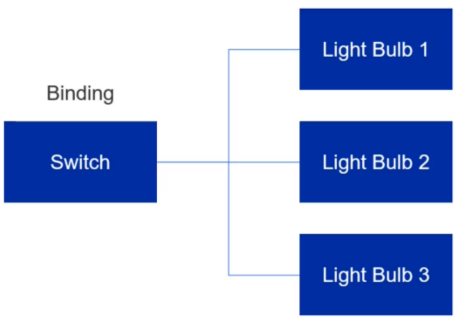


- Label
  - “room”: ”bedroom”

Endpoint

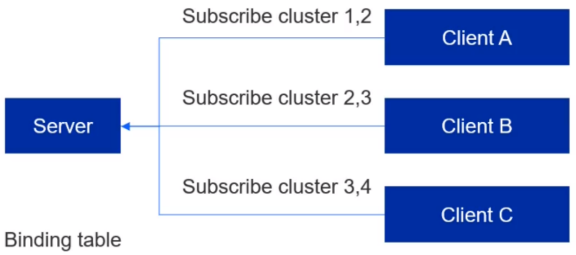
System Model (6/8) - Binding Cluster

- **Attribute**
  - Binding – it's a list of binding target
- **Binding target**
  - Fabric Index
  - Node / Group
  - Endpoint -- the remote endpoint
  - Cluster
- **The binding target could be a single endpoint or a group**



System Model (7/8) - Proxy

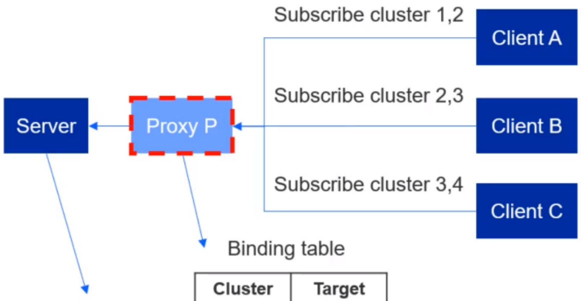
Subscribe with no proxy



Binding table

Cluster	Target
1	A
2	A
2	B
3	B
3	C
4	C

Subscribe with proxy



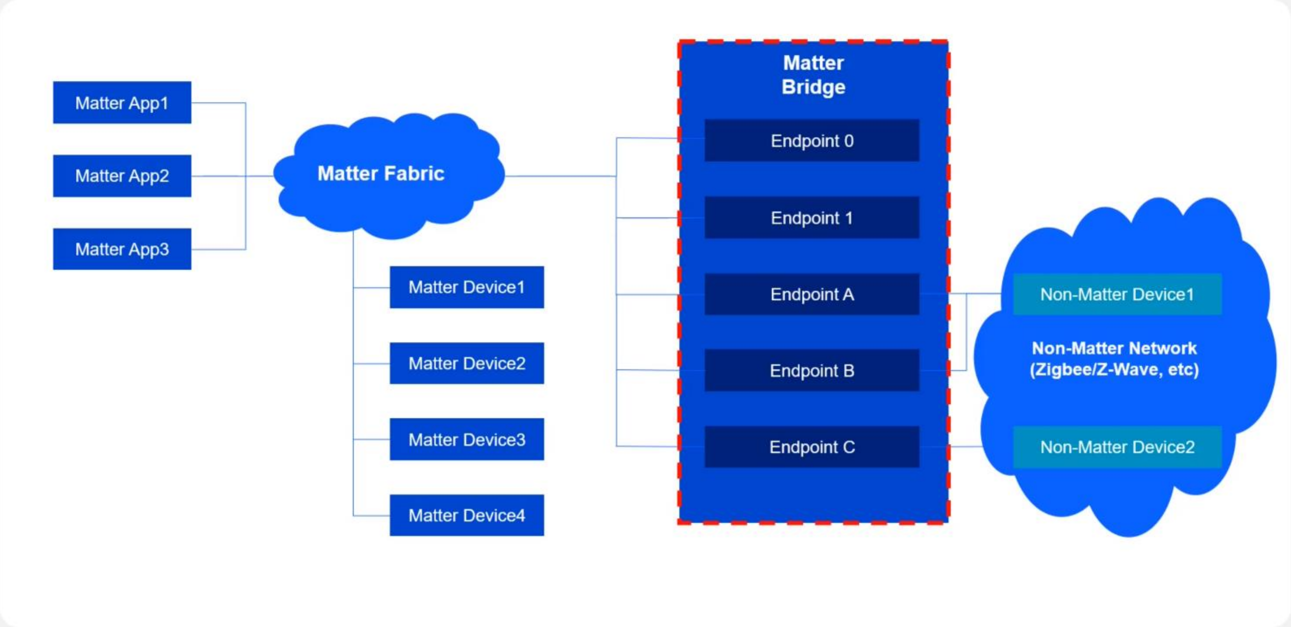
Binding table

Cluster	Target
1	P
2	P
3	P
4	P

Binding table

Cluster	Target
1	A
2	A
2	B
3	B
3	C
4	C

System Model (8/8) - Bridge for Non-Matter Devices



Matter App1

Matter App2

Matter App3

Matter Fabric

Matter Device1

Matter Device2

Matter Device3

Matter Device4

Matter Bridge

Endpoint 0

Endpoint 1

Endpoint A

Endpoint B

Endpoint C

Non-Matter Device1

Non-Matter Network (Zigbee/Z-Wave, etc)

Non-Matter Device2

Security (1/4) - Principles

- **No anonymous joining**
  - Always requires “proof of ownership” (i.e. a device specific Passcode)
- **Device Attestation**
  - Every Device has unique identity that is authenticated by the manufacturer and verified through the CSA as a certified device
- **Operational Credentials**
  - When commissioned onto a Matter network every device is given unique operational credentials after verifying their manufacturer credentials
- **Network credentials are given only *after* device authentication**
  - WiFi network key or Thread Master Key are not given until device’s certificate is verified
- **Open standard and open-source software**
  - Open to third parties vetting the claims by examining the standard and auditing the source code

Security (2/4) - Cryptographic Primitives

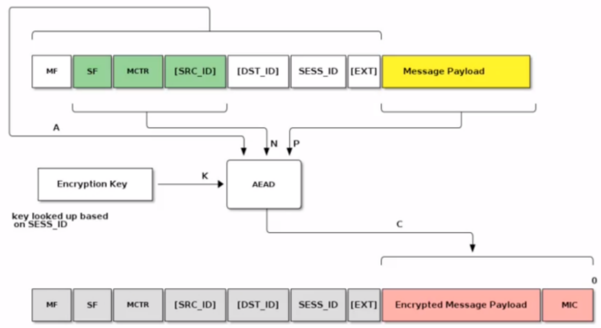
- SHA-256 is the hash algorithm
- HMAC-SHA-256 for message authentication
- NIST P-256 as public key ECC curve
- AES-CCM using 128-bit keys for message encryption

Security (3/4) - Cryptographic Functionality

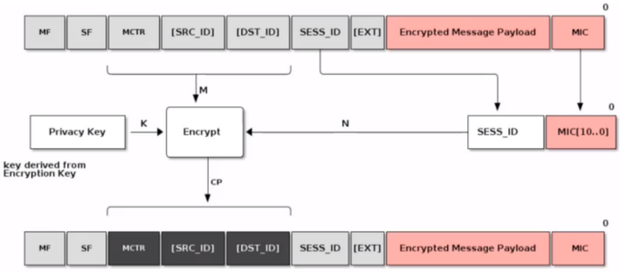
Certificates	Operational Identity	PASE Password authenticated session establishment	CASE Certificate authenticated session establishment
• Natively uses a CHIP TLV format but can convert to/from X.509 format	• All devices are given an operational certificate to prove their authorization on the Matter network (fabric) and securely identify them	• Used during initial setup to verify possession of the passcode by both the commissioner and joining device	• Used during normal operation between controller and device to validate that both are part of the Matter network

Security (4/4) - Message Security

- **Confidentiality**
  - Message payload is encrypted by the **encryption key**

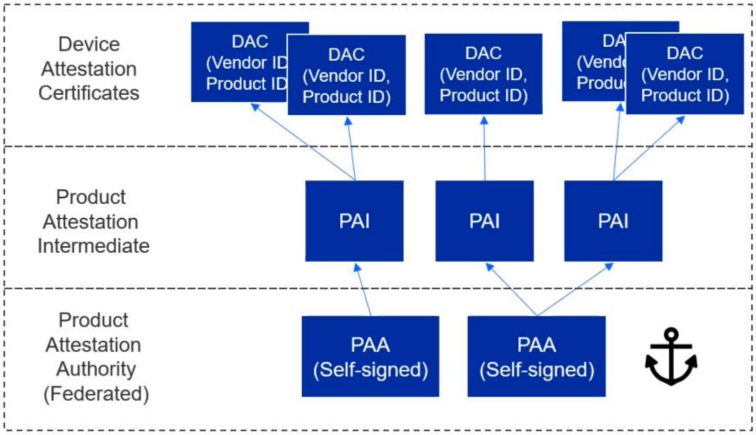


- **Privacy**
  - Addresses are encrypted by the **privacy key**



Device Attestation (1/2) - Device Certificates

- **Every device has a unique certificate that is signed by the manufacturer**
- **The hierarchy allows for a 3-level tier**
- **No single root CA across all devices**
- **During commissioning the device is challenged to prove possession of associated private key**
- **Certificate can be validated against the Distributed Compliance Ledger to verify device certification status**



Device Attestation Certificates

DAC (Vendor ID Product ID

DAC (Vendor ID, Product ID)

DAC (Vendor ID, Product ID)

DA (Vend Produ

DAC (Vendor ID, Product ID)

Product Attestation Intermediate

PAI

PAI

PAI

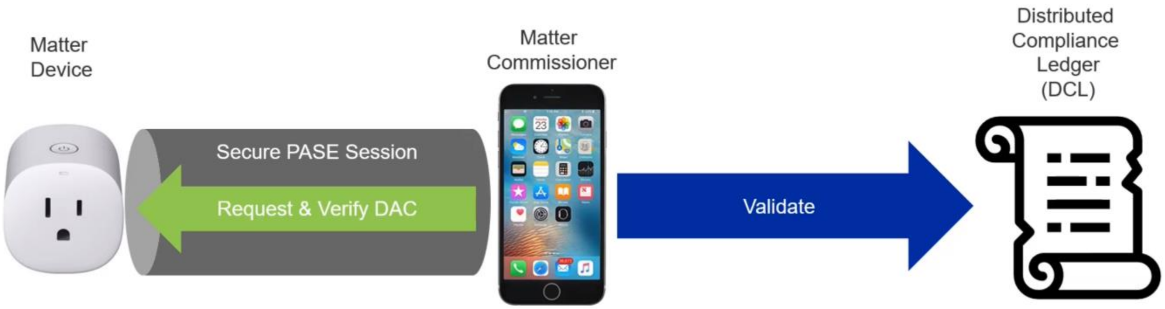
Product Attestation Authority (Federated)

PAA (Self-signed)

PAA (Self-signed)

⚓

Device Attestation (2/2) - Checking the Compliance Ledger

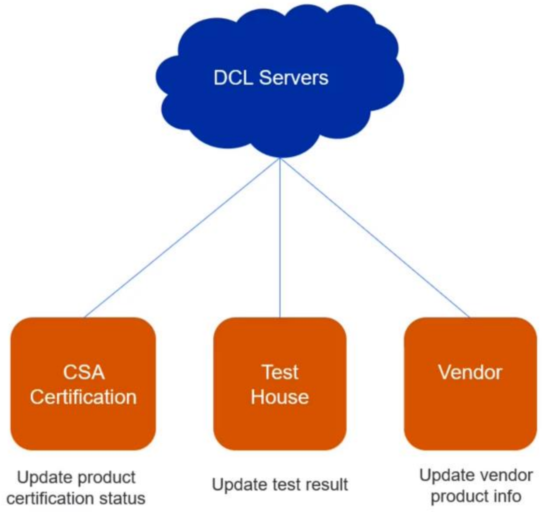


DAC is retrieved & verified prior to device joining the Thread or WiFi network.

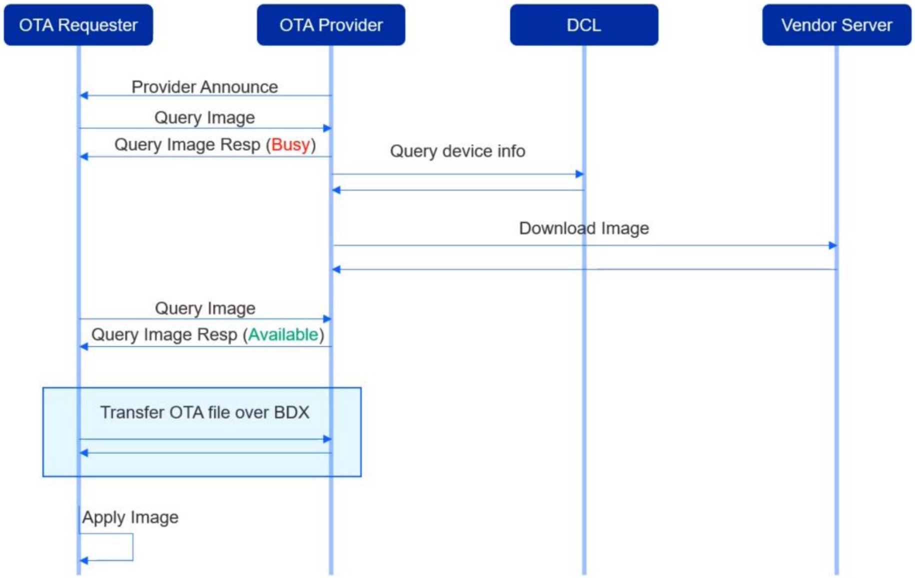
Commissioner issues a challenge to the device to prove it possesses the associated Private Key

Distributed Compliance Ledger

- **DCL**
  - Distributed database of all certified products
    - Certification status
    - Product name / description / firmware URL
    - Product certificates
- **Read from DCL is public**
- **Write to DCL is restricted**
  - CSA Certification role
  - Test House role
  - Vendor role



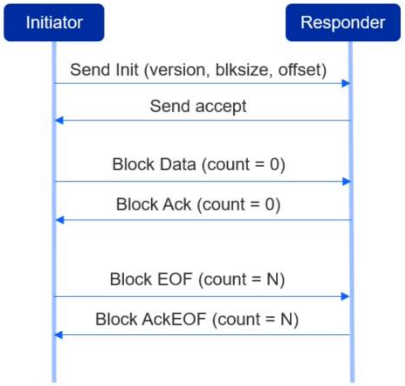
OTA (1/2) - Sequence Flow



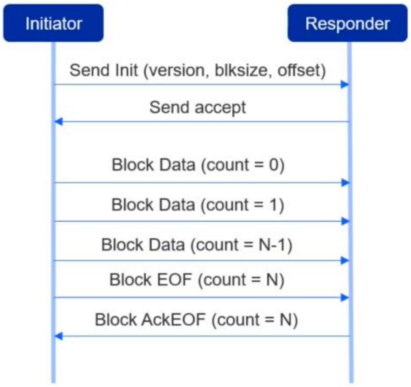
OTA (2/2) – BDX: Bulk Data Exchange Protocol

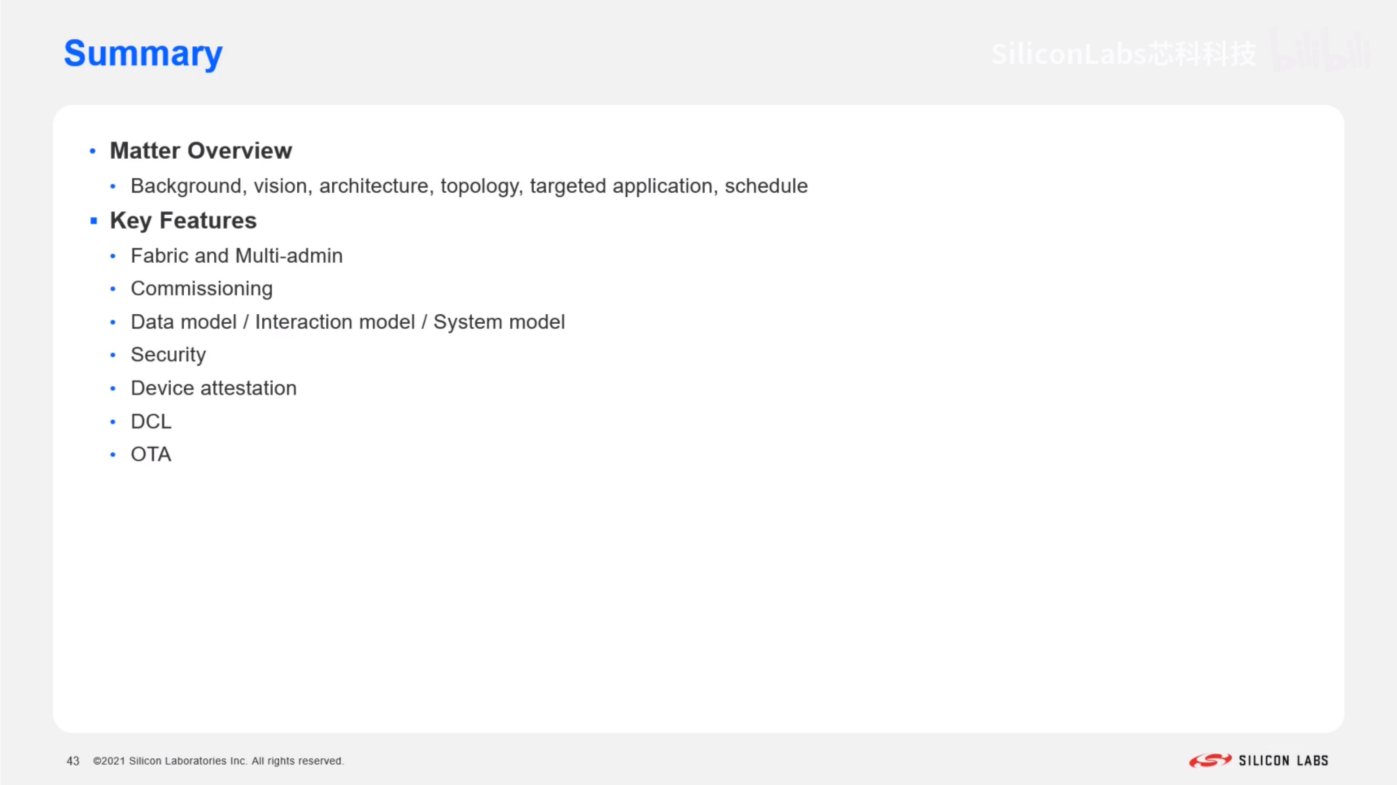
BDX is a file transfer protocol used in Matter

Synchronous Transfer



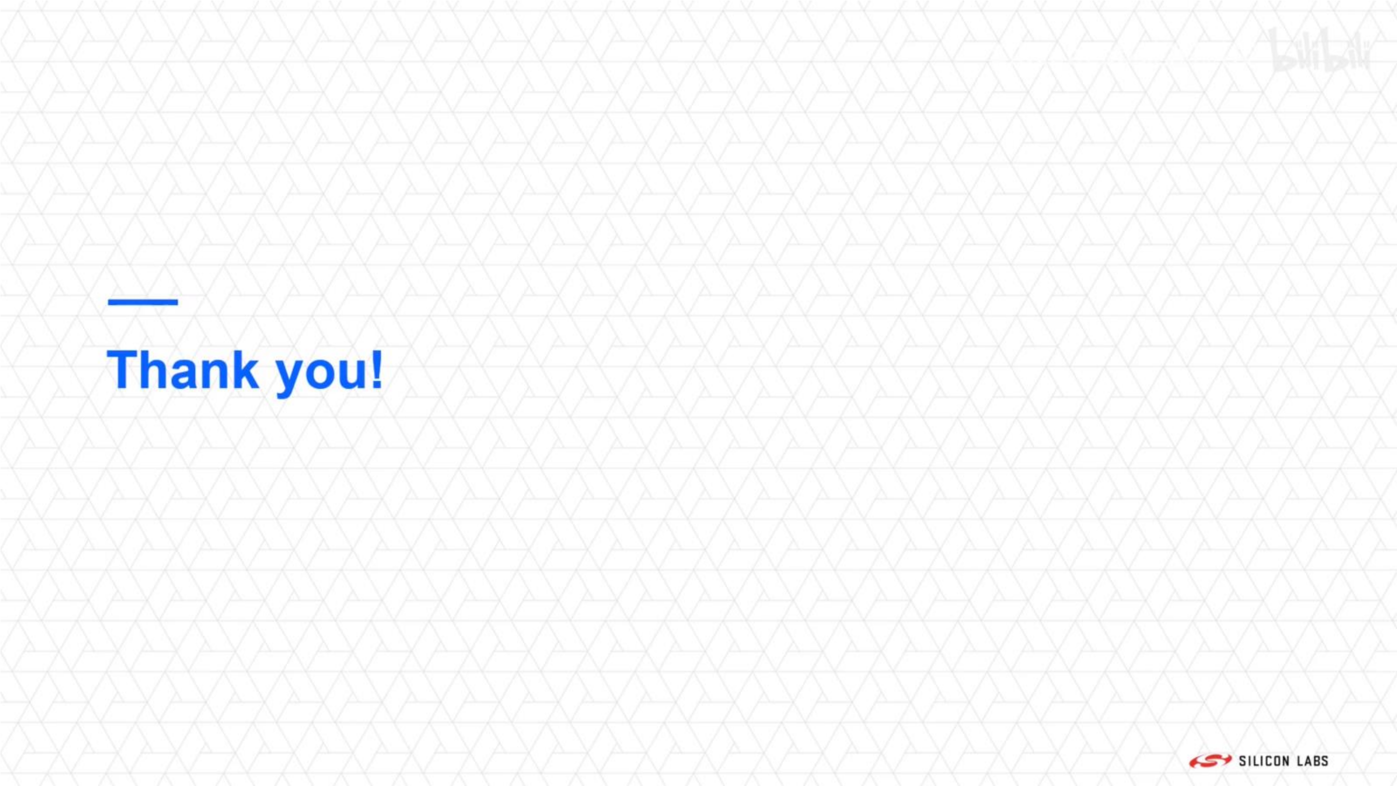
Asynchronous Transfer





Summary

- **Matter Overview**
  - Background, vision, architecture, topology, targeted application, schedule
- **Key Features**
  - Fabric and Multi-admin
  - Commissioning
  - Data model / Interaction model / System model
  - Security
  - Device attestation
  - DCL
  - OTA



Thank you!