



Beyond Local Connectivity: Bridging Web of Things and Matter

Till Langen, 14.03.2024



ERLEBEN, WAS VERBINDET.

Who I am



Till Langen

Dual Student at Deutsche Telekom

Started working in the Smart Home Domain in 2020

Agenda

1. Smart Home at Deutsche Telekom

2. Matter Introduction

- Layers
- Data model
- Endpoint composition
- Matter Bridge

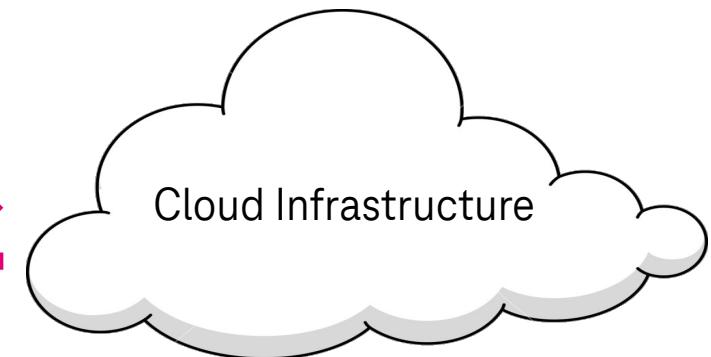
3. Concept

- Creating a Thing Description
- Protocol Binding
- Use of Thing Model

4. Implementation

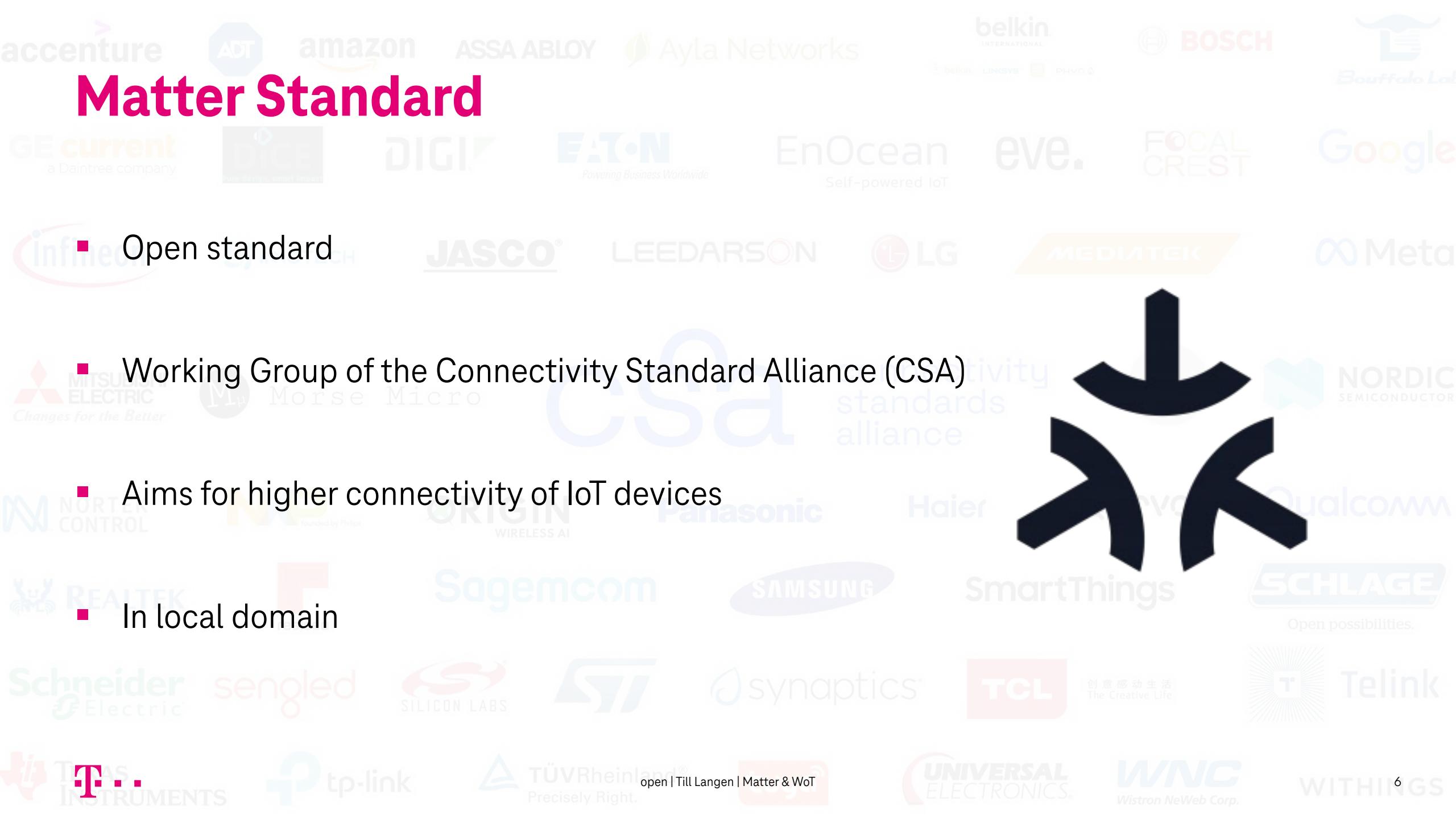


Smart Home at Deutsche Telekom





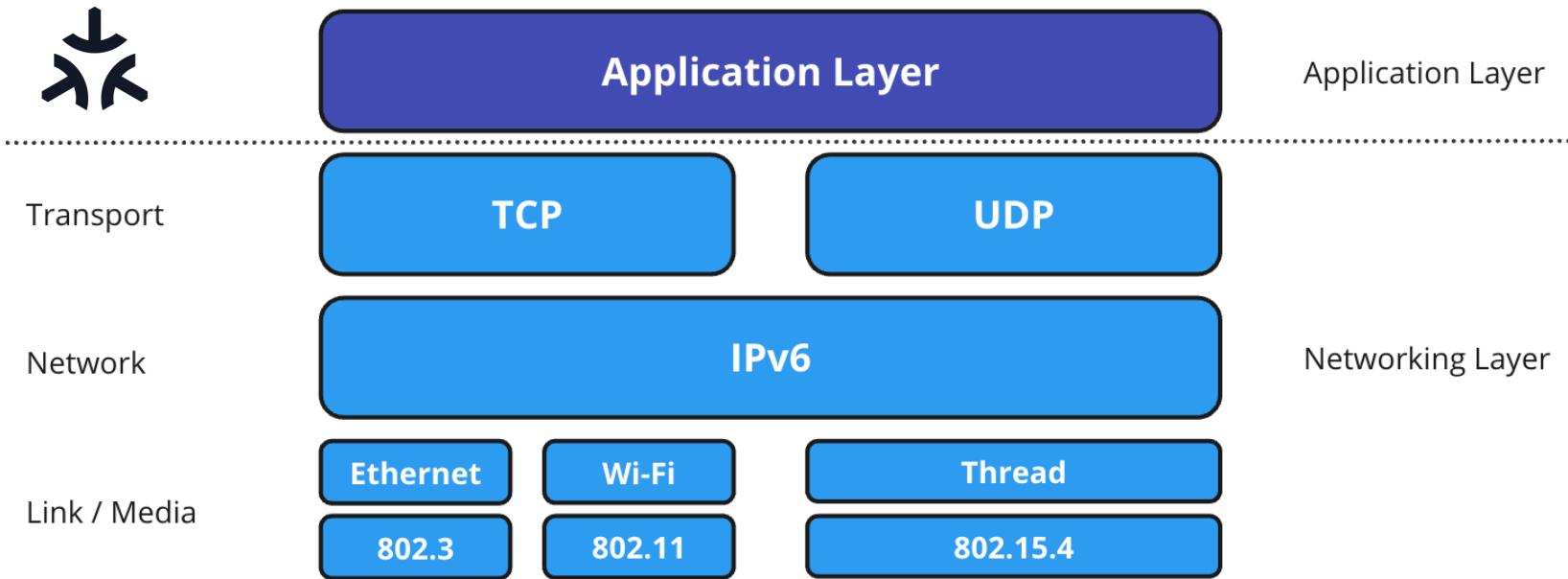
Matter introduction



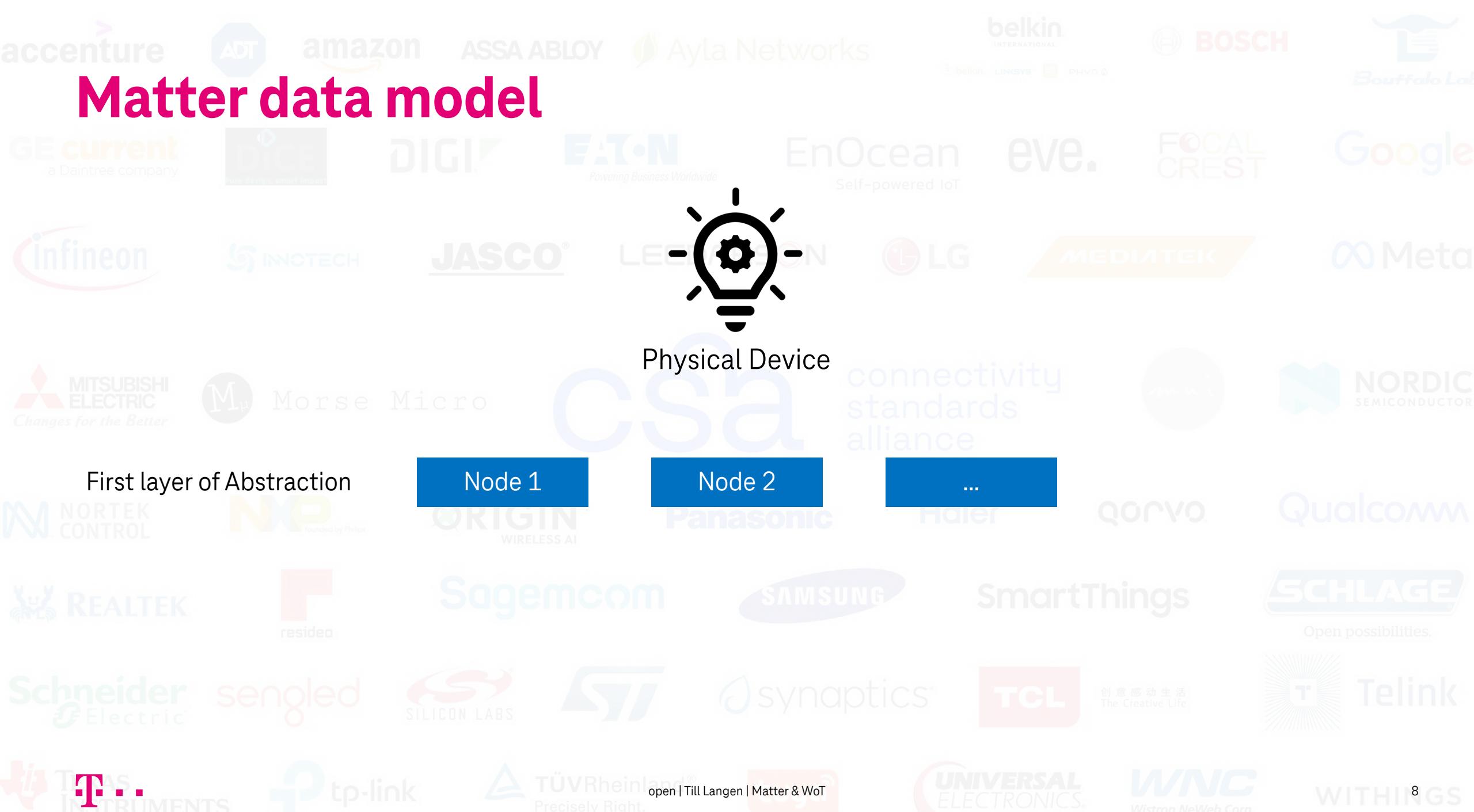
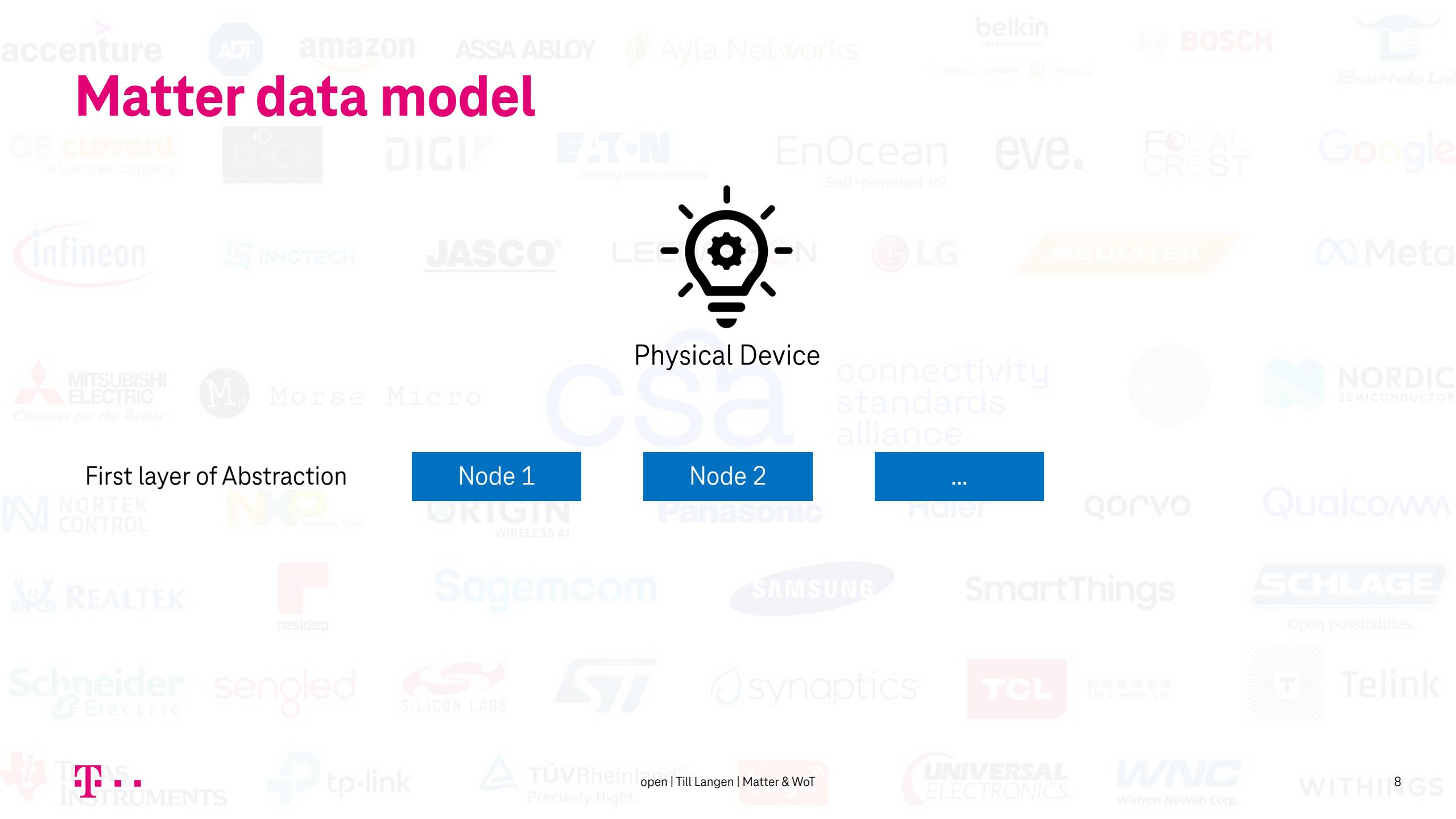
Matter Standard

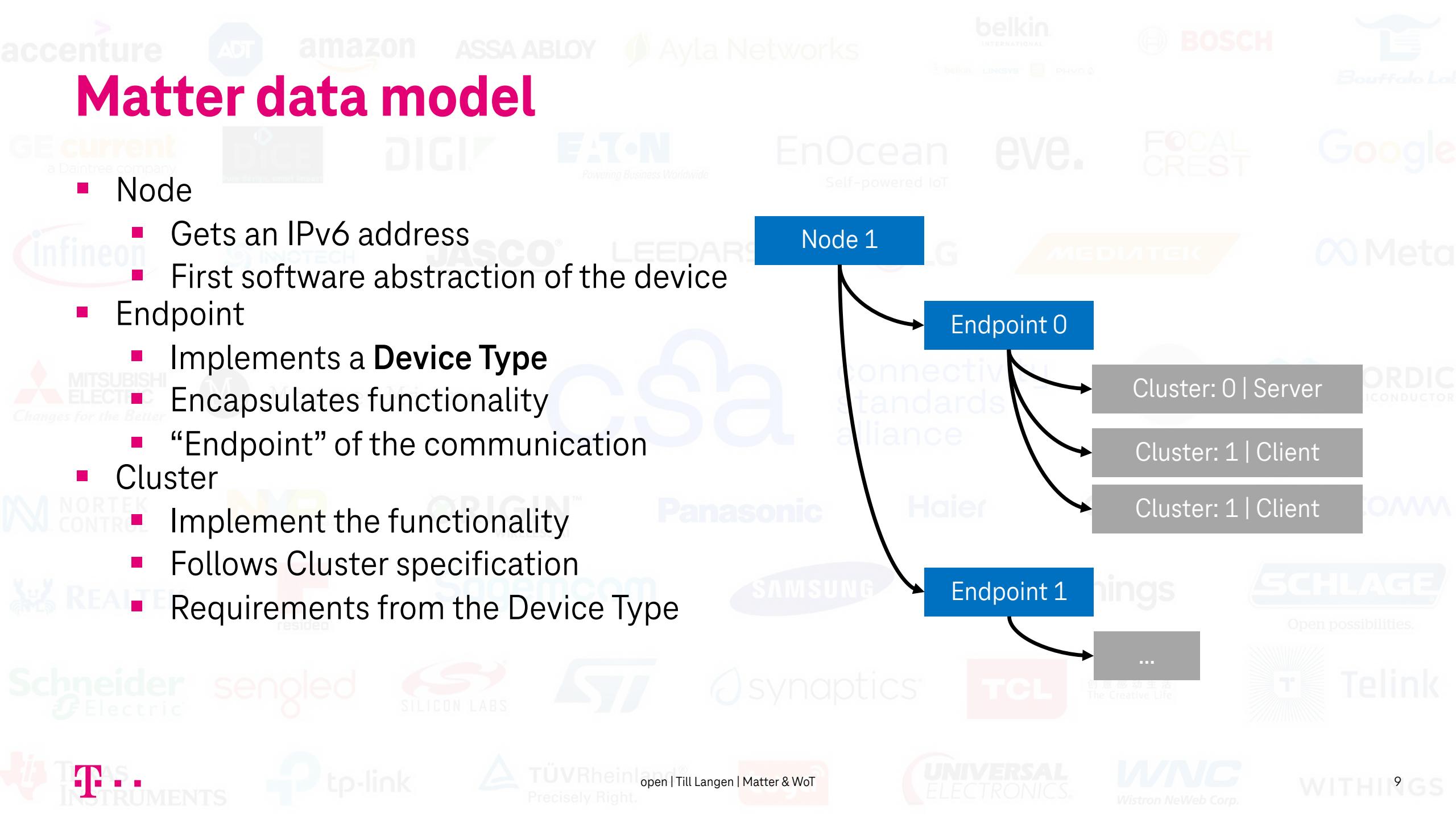
- Open standard
- Working Group of the Connectivity Standard Alliance (CSA)
- Aims for higher connectivity of IoT devices
- In local domain

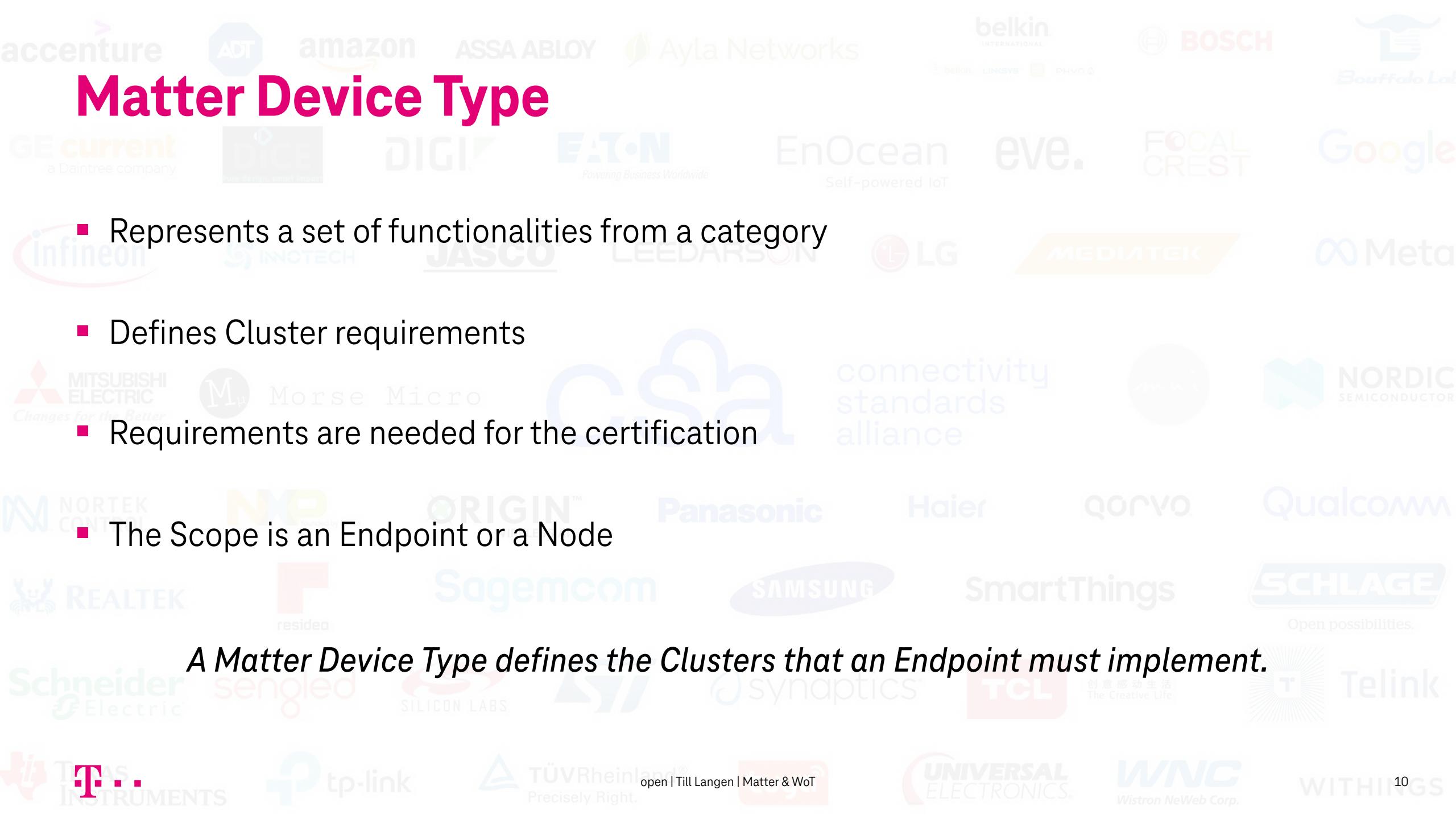
Matter Layer



- Virtual network on Application Layer called ***Fabric***
- May be more than one Fabric



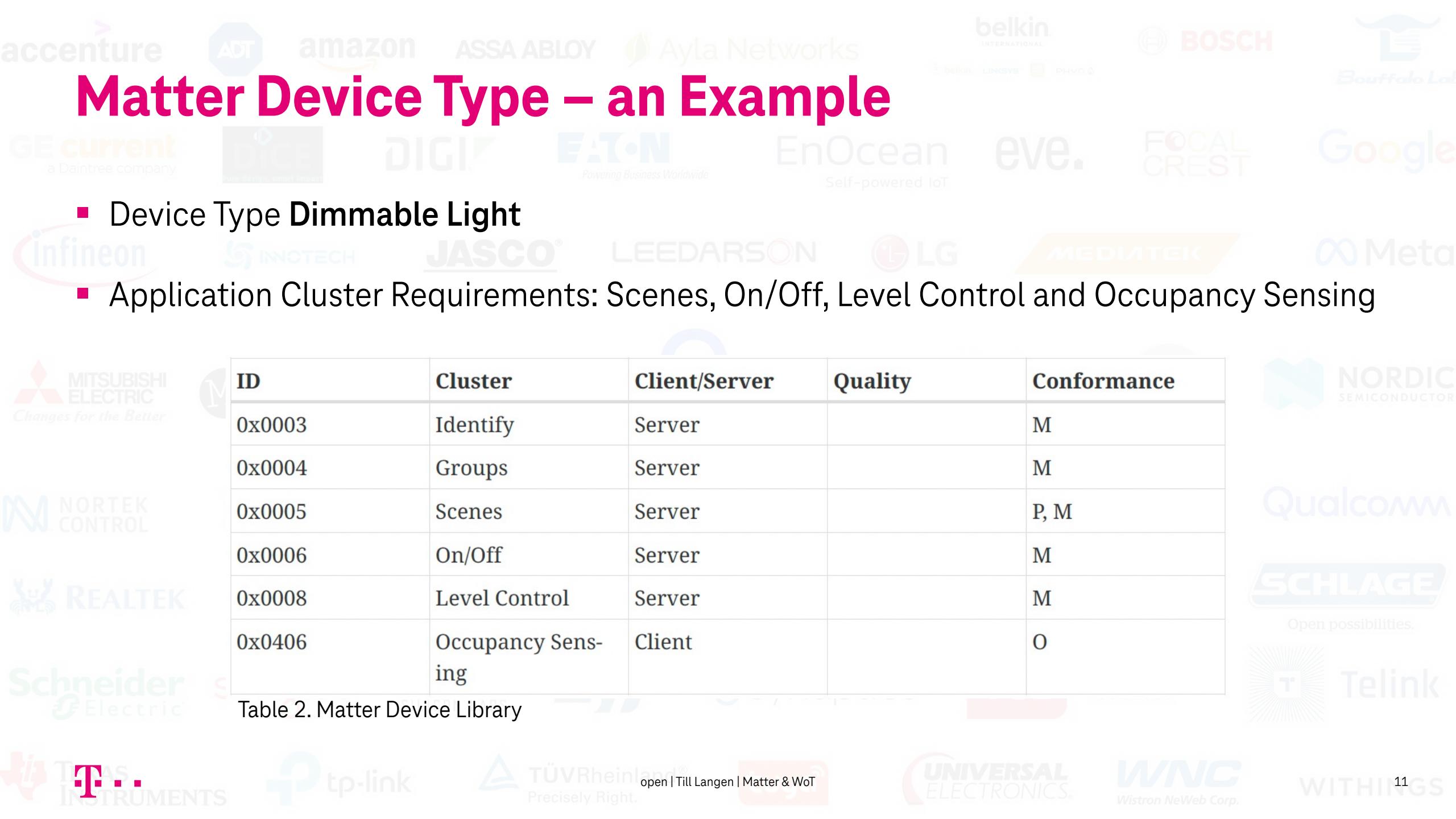




Matter Device Type

- Represents a set of functionalities from a category
- Defines Cluster requirements
- Requirements are needed for the certification
- The Scope is an Endpoint or a Node

A Matter Device Type defines the Clusters that an Endpoint must implement.

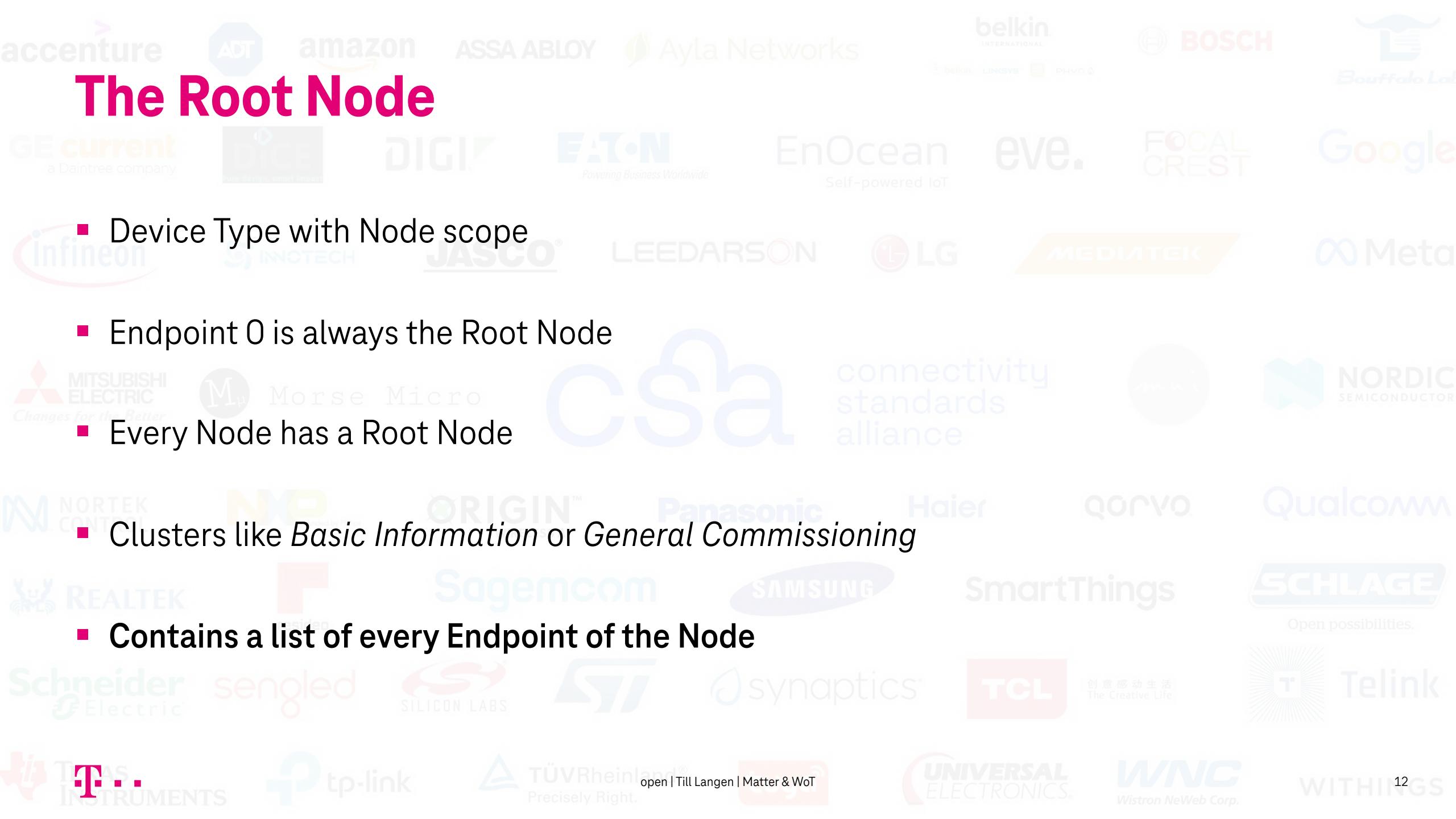


Matter Device Type – an Example

- Device Type **Dimmable Light**
- Application Cluster Requirements: Scenes, On/Off, Level Control and Occupancy Sensing

ID	Cluster	Client/Server	Quality	Conformance
0x0003	Identify	Server		M
0x0004	Groups	Server		M
0x0005	Scenes	Server		P, M
0x0006	On/Off	Server		M
0x0008	Level Control	Server		M
0x0406	Occupancy Sensing	Client		O

Table 2. Matter Device Library



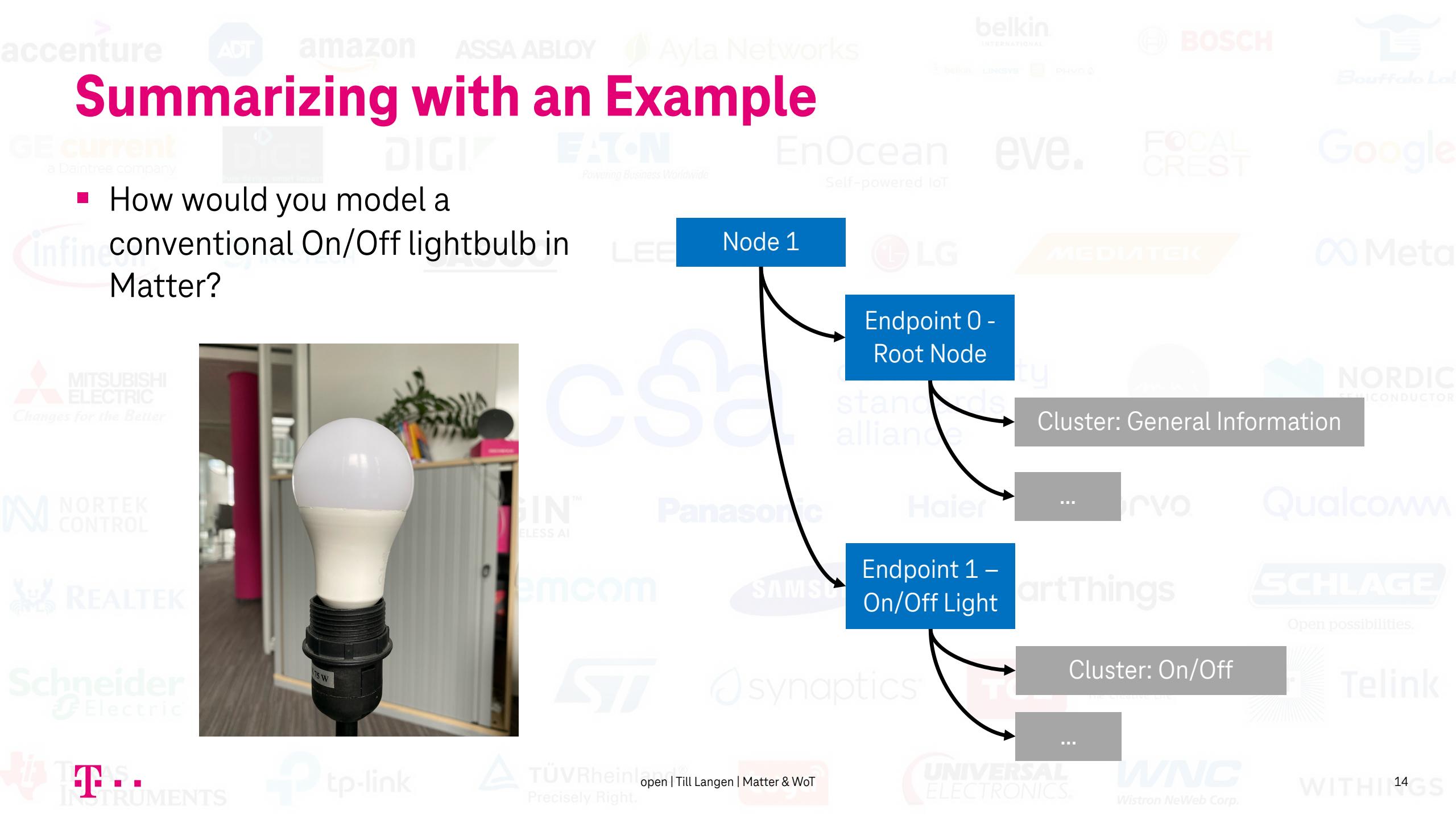
The Root Node

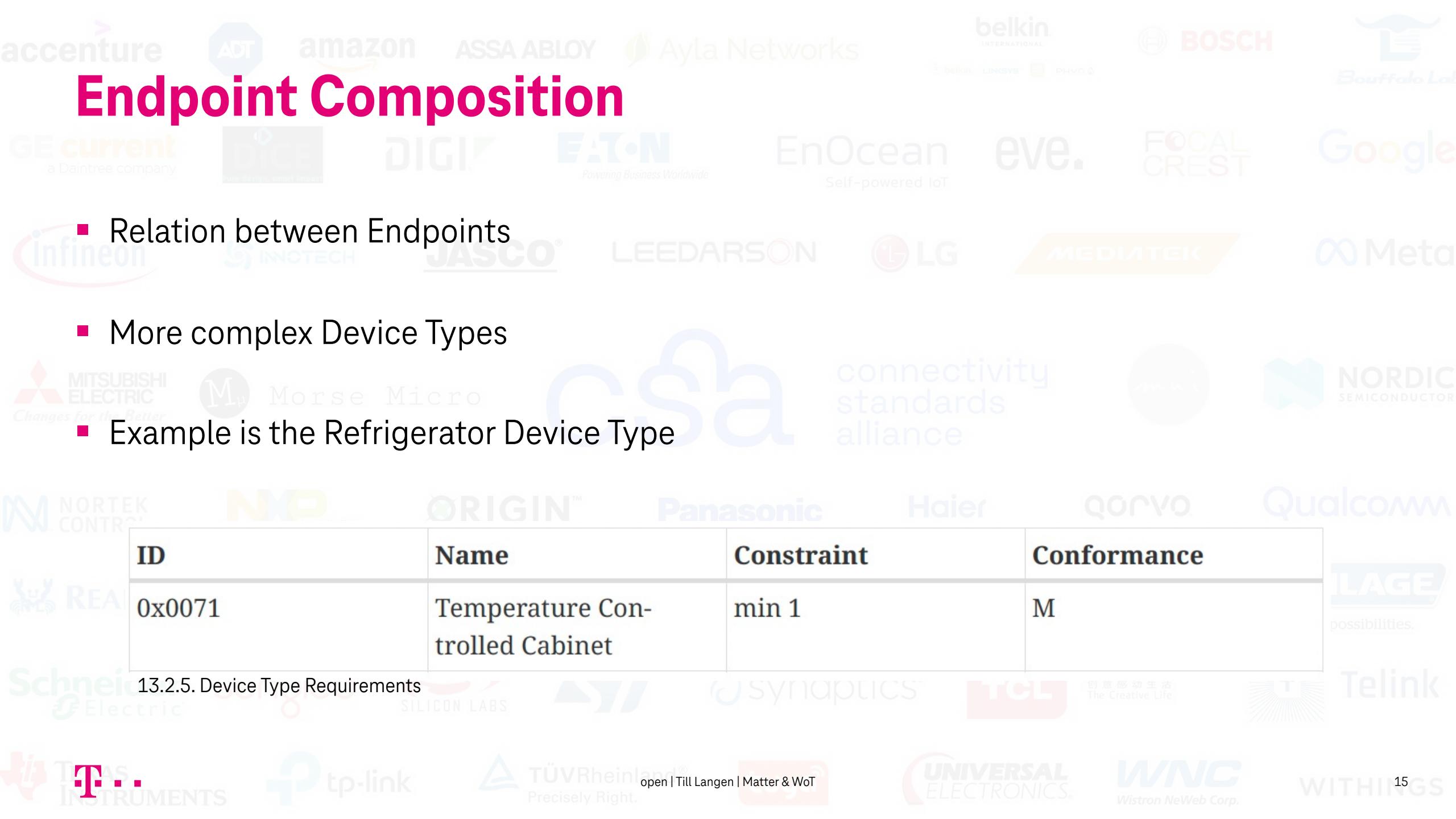
- Device Type with Node scope
- Endpoint 0 is always the Root Node
- Every Node has a Root Node
- Clusters like *Basic Information* or *General Commissioning*
- Contains a list of every Endpoint of the Node



Matter Cluster

- Implements a functionality
- Is specified by the Cluster specification
- Consists of **Attributes, Commands and Events**
- Can have cluster specific response codes
- Examples: *On/Off, Level Control, Color Control or Door Lock*



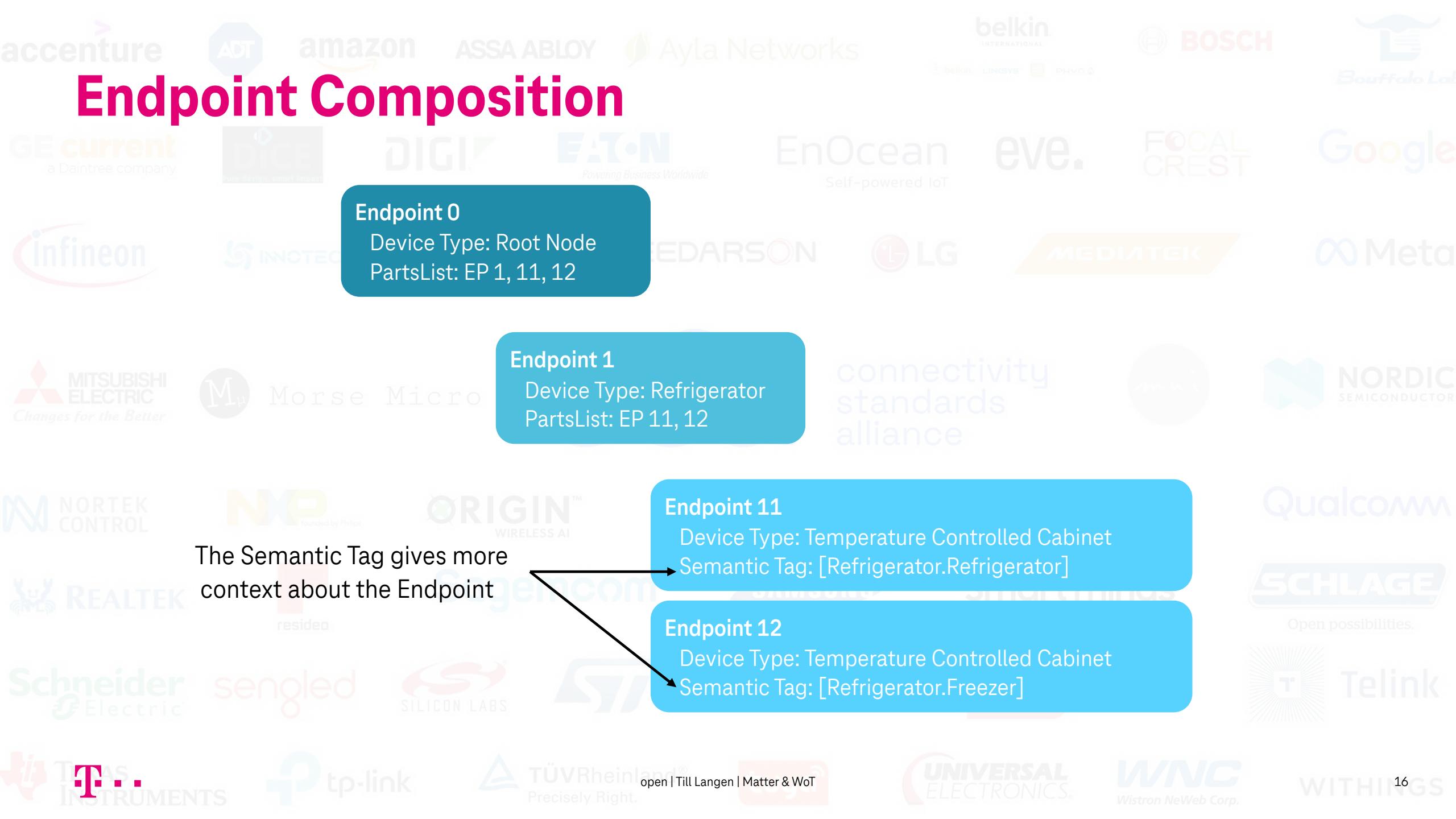


Endpoint Composition

- Relation between Endpoints
- More complex Device Types
- Example is the Refrigerator Device Type

ID	Name	Constraint	Conformance
0x0071	Temperature Controlled Cabinet	min 1	M

13.2.5. Device Type Requirements



accenture

ADT

amazon

ASSA ABLOY

Ayla Networks

belkin INTERNATIONAL

BOSCH

Buffalo Lab

GE current a Daintree company

DIGI

EATON Powering Business Worldwide

EnOcean Self-powered IoT

eve.

FOCAL CREST

Google

infineon

INNTECH

LG

MEDIATEK

MITSUBISHI ELECTRIC Changes for the Better

Morse Micro

NORDIC SEMICONDUCTOR

NORTEK CONTROL

ORIGIN WIRELESS AI

Qualcomm

REALTEK

SCHLAGE Open possibilities.

Schneider Electric

sengled

SILICON LABS

STMicroelectronics

TÜV Rheinland Precisely Right.

UNIVERSAL ELECTRONICS

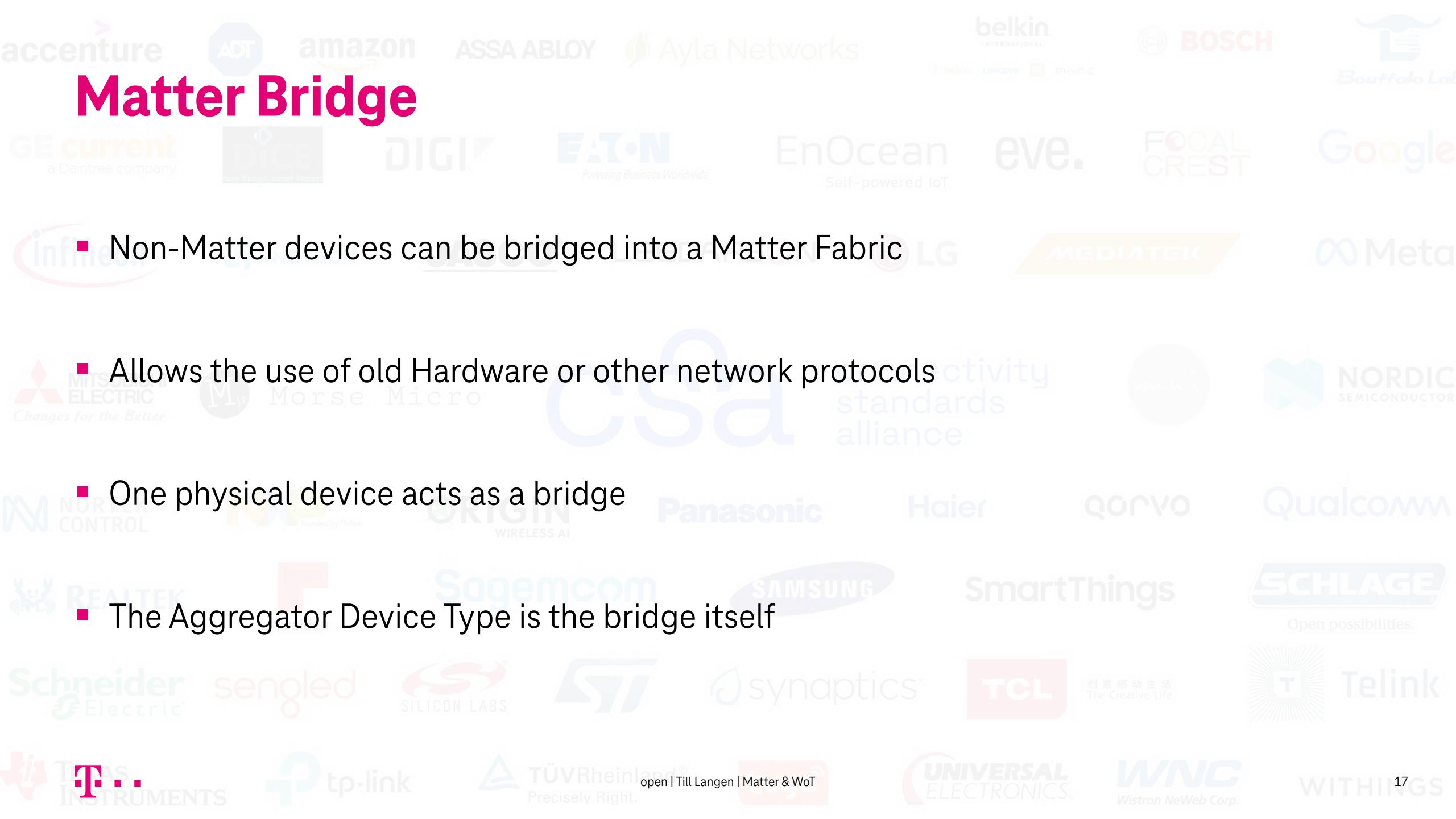
VNC Wistron NeWeb Corp.

WITHINGS

TAS INSTRUMENTS

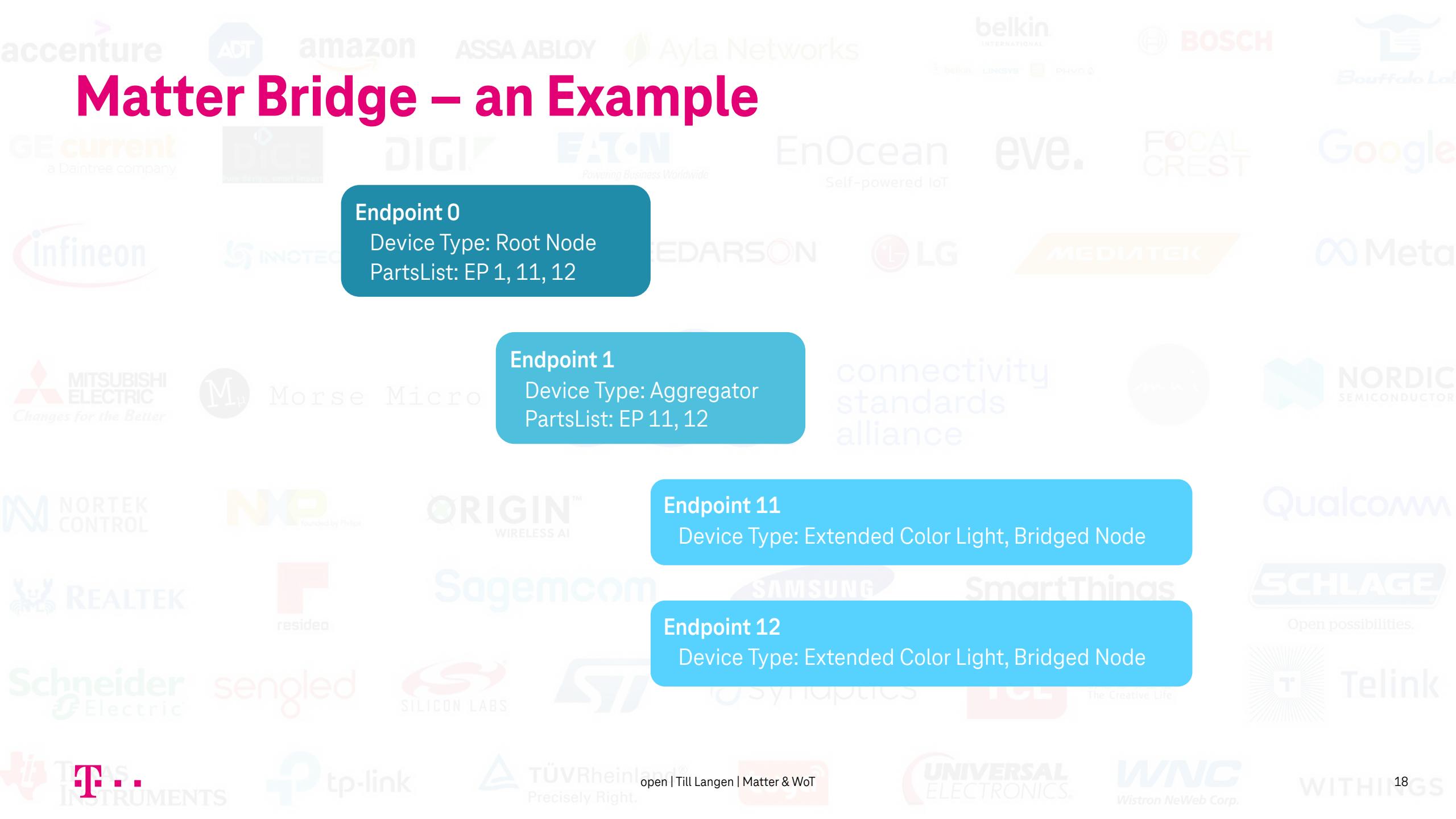
tp-link

16



Matter Bridge

- Non-Matter devices can be bridged into a Matter Fabric
- Allows the use of old Hardware or other network protocols
- One physical device acts as a bridge
- The Aggregator Device Type is the bridge itself



Matter Bridge – an Example

Endpoint 0

Device Type: Root Node
PartsList: EP 1, 11, 12

Endpoint 1

Device Type: Aggregator
PartsList: EP 11, 12

Endpoint 11

Device Type: Extended Color Light, Bridged Node

Endpoint 12

Device Type: Extended Color Light, Bridged Node

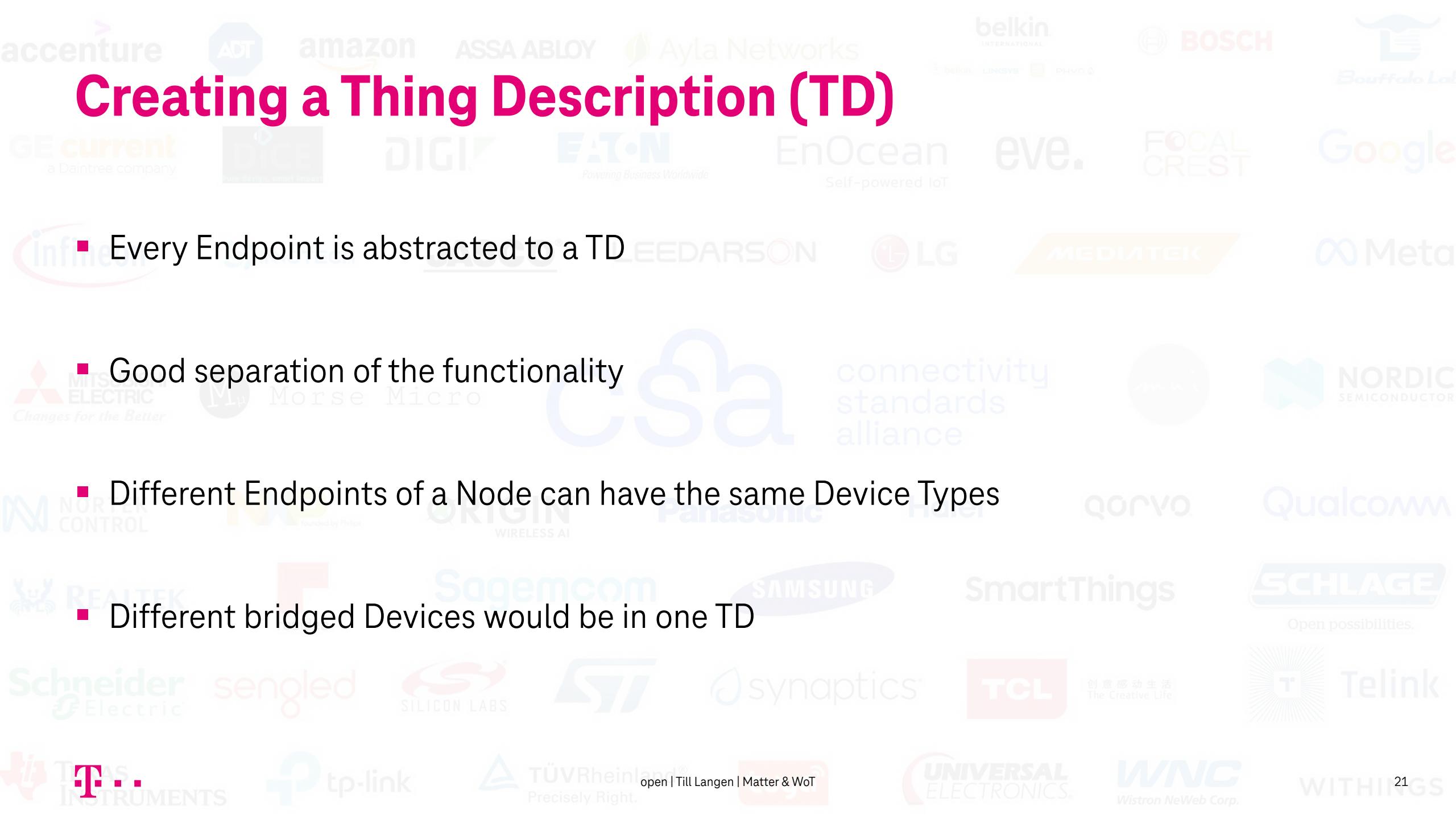


Matter access control

- Defines the access rights read, write and invoke
- For every characteristic (Attribute, Command and Event)
- Operate-, Manage- and Administer-Privilege
- Imply each other (e.g. invoke needs a Operate-Privilege)



Concept



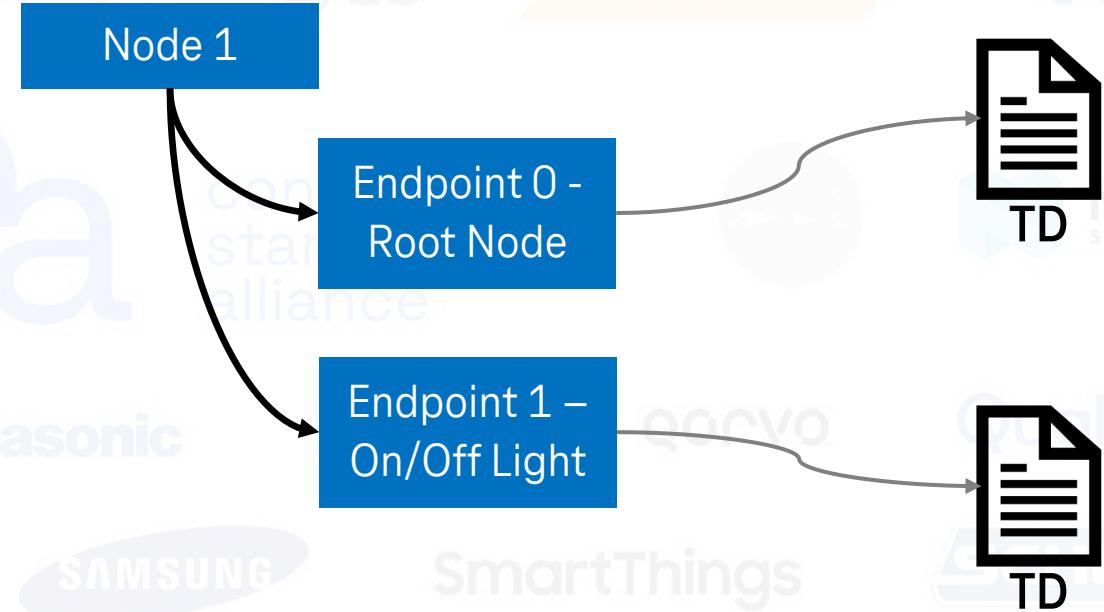
Creating a Thing Description (TD)

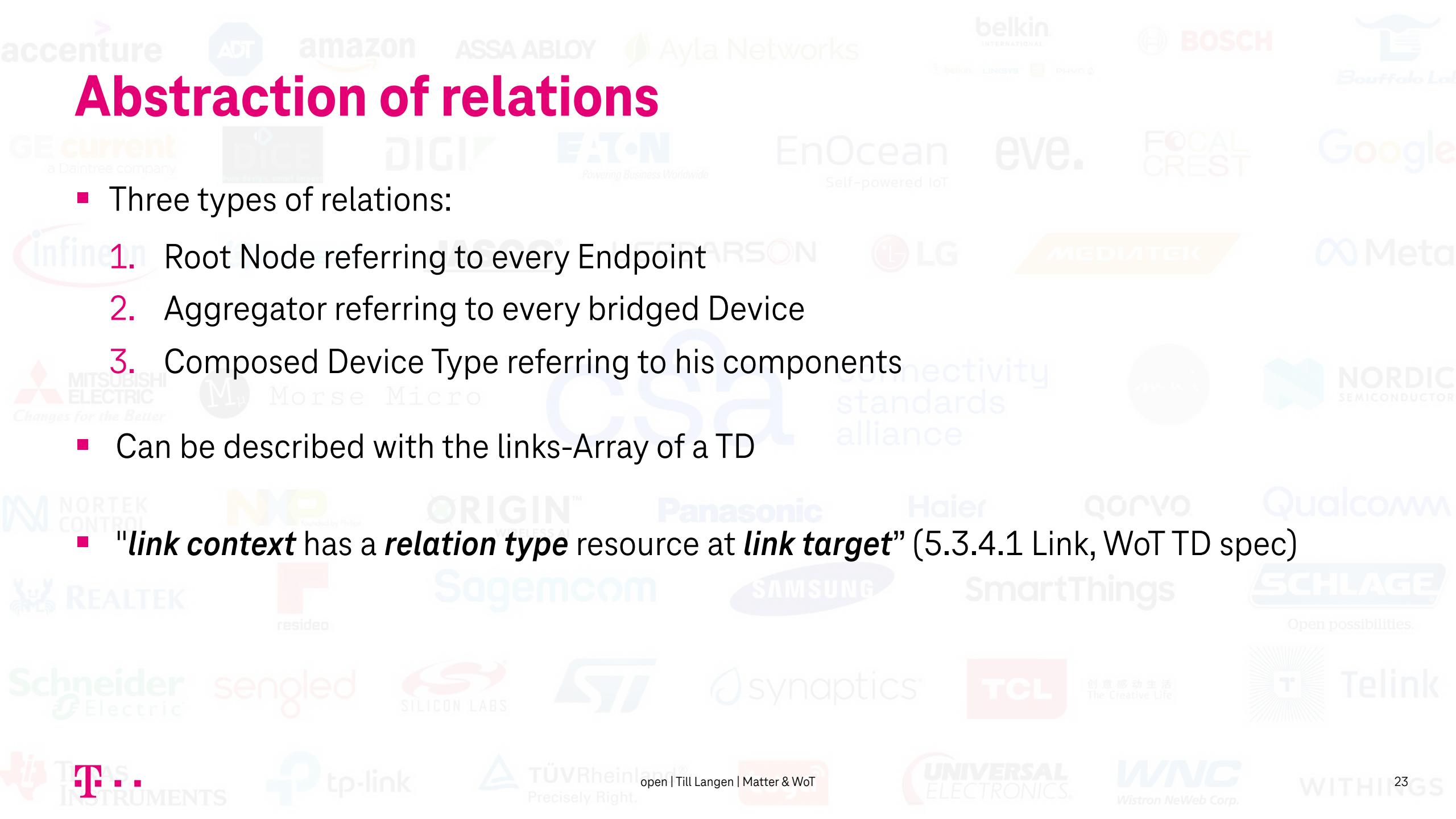
- Every Endpoint is abstracted to a TD
- Good separation of the functionality
- Different Endpoints of a Node can have the same Device Types
- Different bridged Devices would be in one TD



Creating a Thing Description – an Example

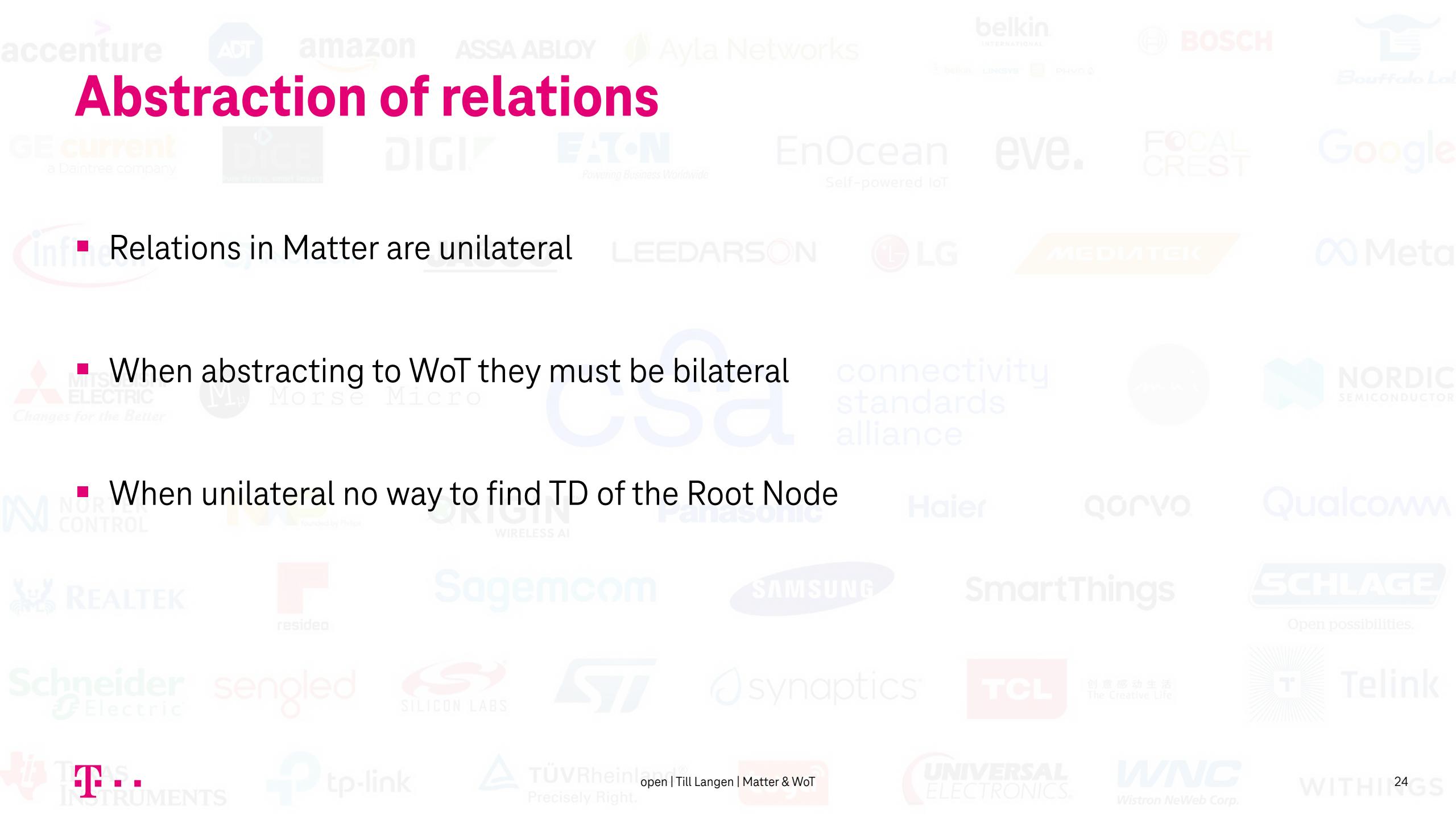
- Creating two TDs
- One for the Root Node
- One for the On/Off Light
- Leads to loss of the belonging to a node





Abstraction of relations

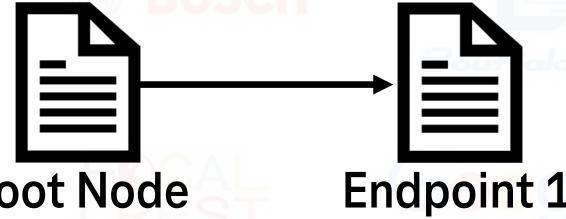
- Three types of relations:
 1. Root Node referring to every Endpoint
 2. Aggregator referring to every bridged Device
 3. Composed Device Type referring to his components
- Can be described with the links-Array of a TD
- "**link context** has a **relation type** resource at **link target**" (5.3.4.1 Link, WoT TD spec)



Abstraction of relations

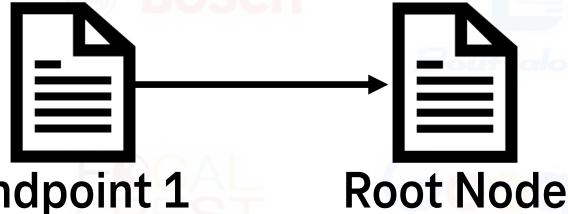
- Relations in Matter are unilateral
- When abstracting to WoT they must be bilateral
- When unilateral no way to find TD of the Root Node

Relation type for the Root Node



- Root Node relation to the Endpoint using the Parts List Attribute:

```
{  
    "@type": "RootNode:0x0016",  
    "links": [  
        {  
            "rel": "matterDescriptorCluster:0x0003",  
            "href": "http://endpoint1"  
        },  
        {  
            "rel": "matterDescriptorCluster:0x0003",  
            "href": "http://endpoint2"  
        }  
    ],  
    ...  
}
```



Relation type for the Root Node

- Endpoint Relation to the Root Node using a Link Relation from the iana
- Describedby: “Refers to a resource providing information about the link's context.”

```

{
  "links": [
    {
      "rel": "describedby",
      "href": "http://endpoint0"
    }
  ],
  ...
}
  
```

Relation type for the Aggregator



- Aggregator referring to bridged device using Parts List Attribute (same as Root Node)

```
{  
  "links": [  
    {  
      "rel": "matterDescriptorCluster:0x0003",  
      "href": "http://endpoint11"  
    },  
    {  
      "rel": "matterDescriptorCluster:0x0003",  
      "href": "http://endpoint12"  
    },  
    ...  
  ]  
}
```

Relation type for the Aggregator

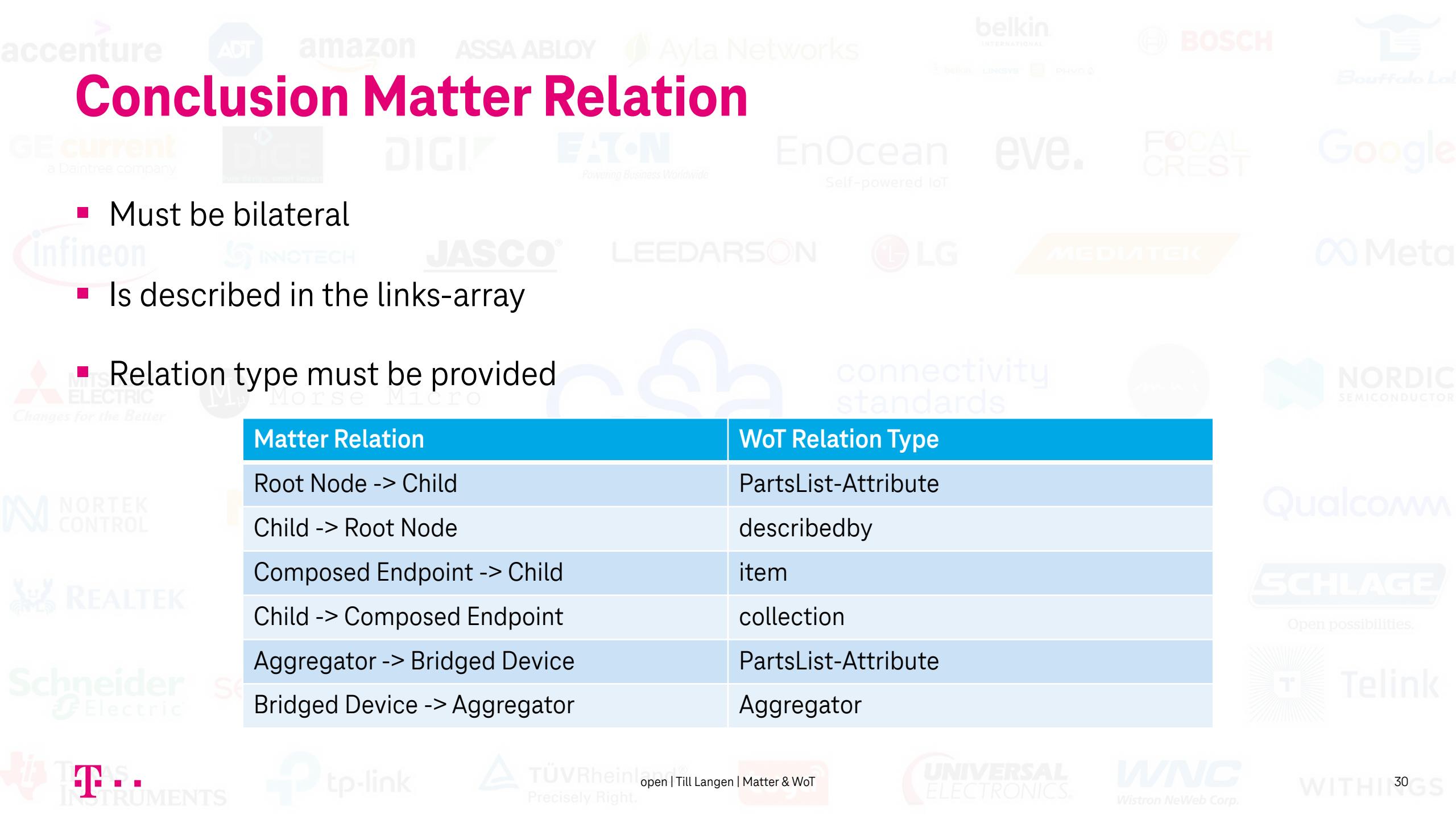


- Bridged device referring to Aggregator
- No matching Link Relation from iana
- Creating new relation type Aggregator

```
{  
  "links": [  
    {  
      "rel": "Aggregator",  
      "href": "http://endpoint21"  
    }  
  ],  
  ...  
}
```

Relation type for composed Device Type

- Using Link Relations from [iana](#)
- **Collection:** points to the resource that is the collection holder
- **Item:** points to a resource that is a part of the collection



Conclusion Matter Relation

- Must be bilateral
- Is described in the links-array
- Relation type must be provided

Matter Relation	WoT Relation Type
Root Node -> Child	PartsList-Attribute
Child -> Root Node	describedby
Composed Endpoint -> Child	item
Child -> Composed Endpoint	collection
Aggregator -> Bridged Device	PartsList-Attribute
Bridged Device -> Aggregator	Aggregator

Abstract Matter access control to WoT

- Properties can make use of the `readOnly` and `writeOnly`
- Matter defined Scopes for all characteristics

```
"securityDefinitions": {  
    "matter_sc": {  
        "scheme": "mattersecurity:SecurityScheme",  
        "scopes": ["view", "operate", "manage", "administer"]  
    }  
},  
"security": ["matter_sc"],  
"actions": {  
    "OnOff:Toggle": {  
        "forms": [{"scopes": ["operate"]}]}  
}
```



Characteristics

- Cluster implement *Attributes, Commands and Events*
- *Attributes abstracted to properties*
- *Commands abstracted to actions*
- *Events abstracted to events*

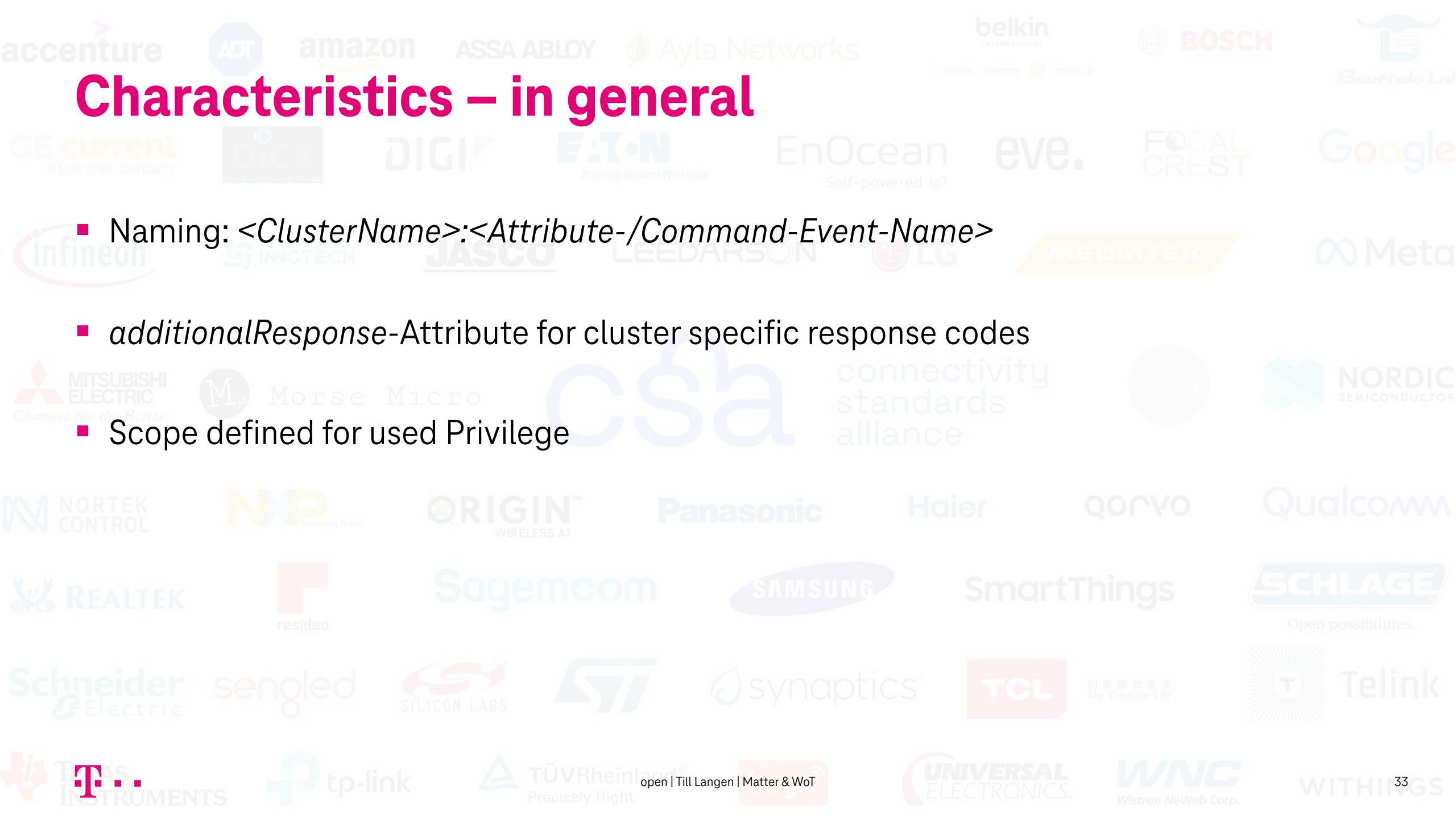
connectivity
standards
alliance

SILICON LABS

TÜV Rheinland[®]
open | Till Langen | Matter & WoT
Precisely Right.

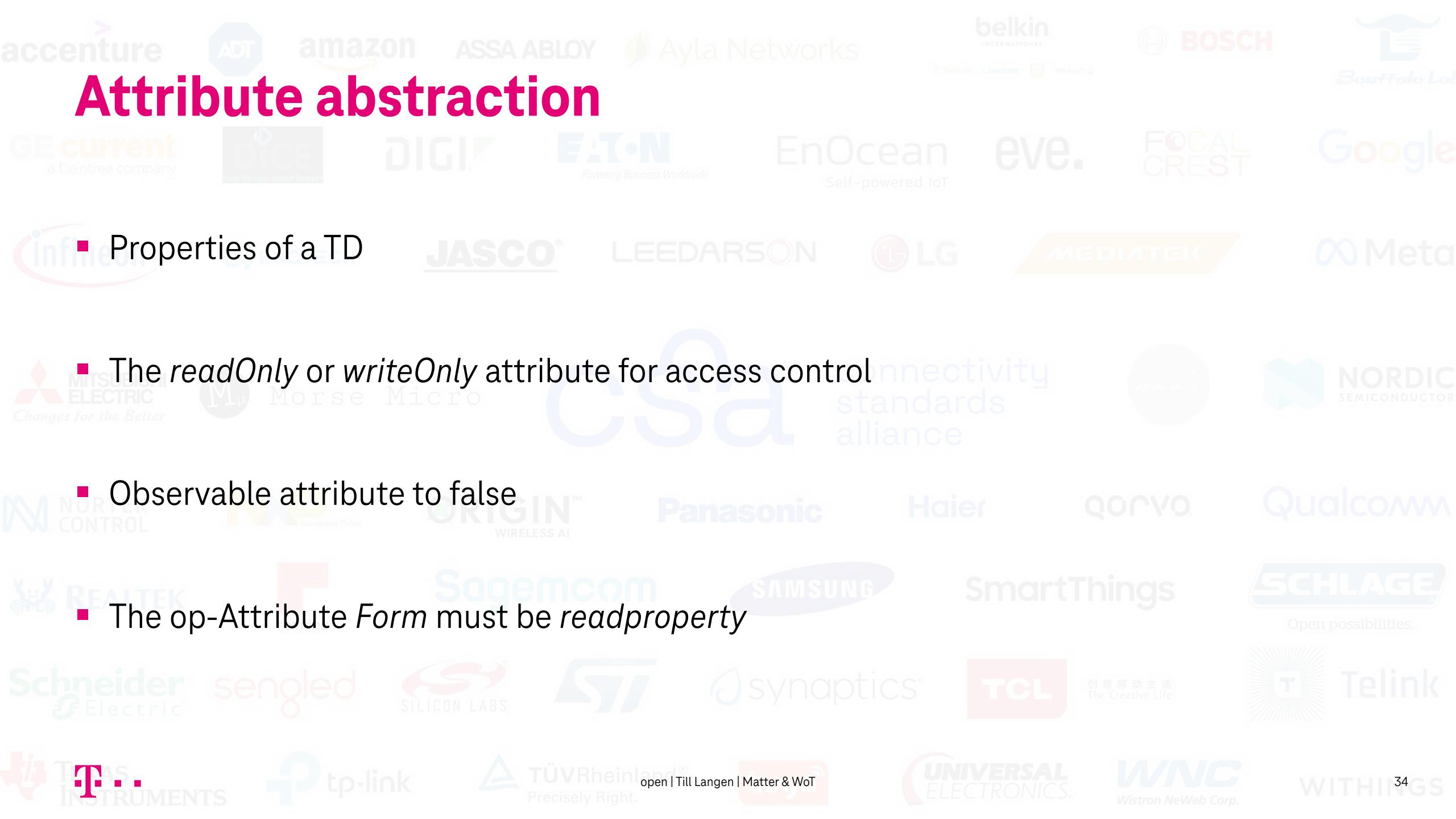
UNIVERSAL
ELECTRONICS

WNC
Wistron NeWeb Corp.



Characteristics – in general

- Naming: <ClusterName>:<Attribute-/Command-Event-Name>
- additionalResponse-Attribute for cluster specific response codes
- Scope defined for used Privilege

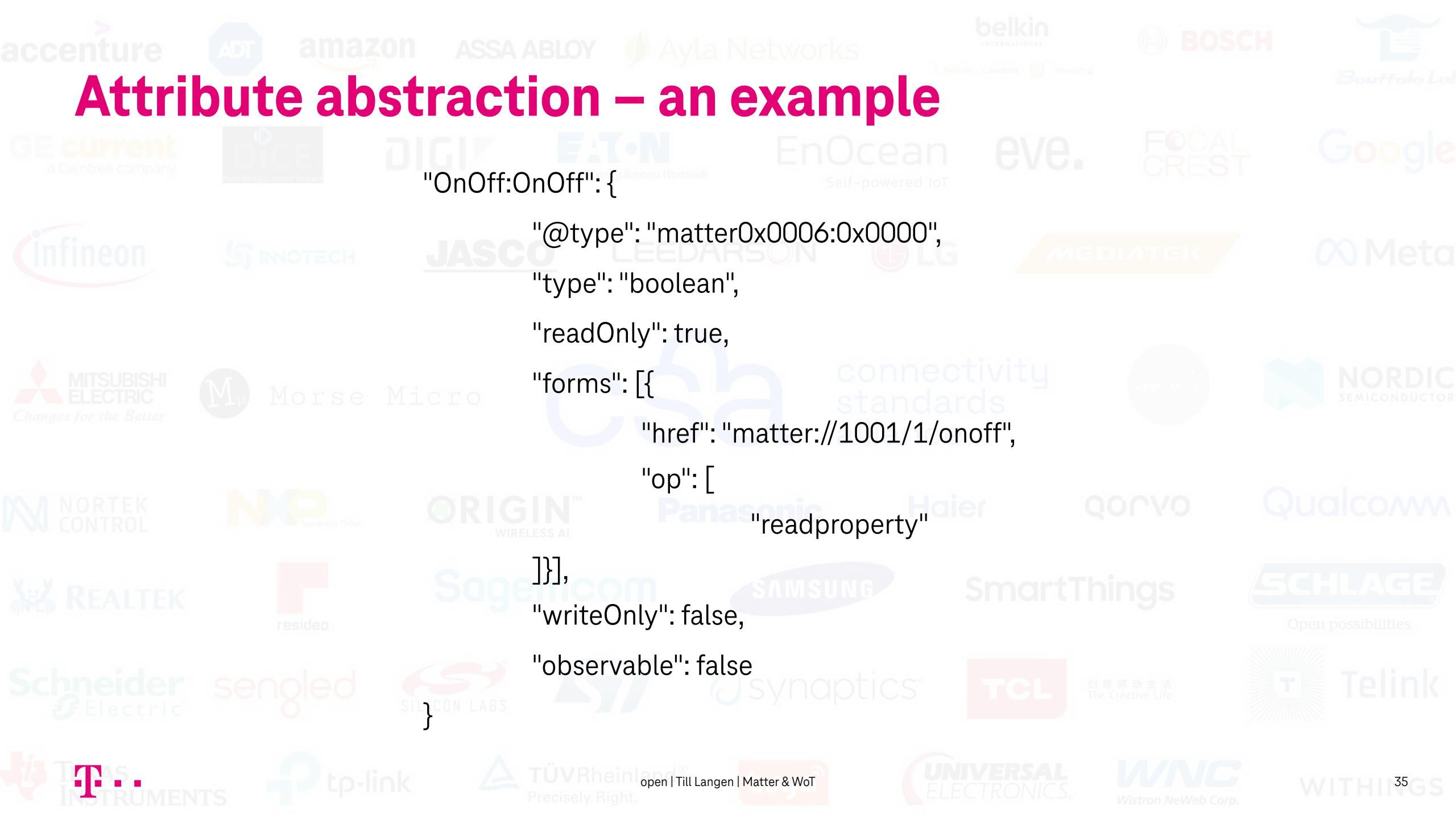


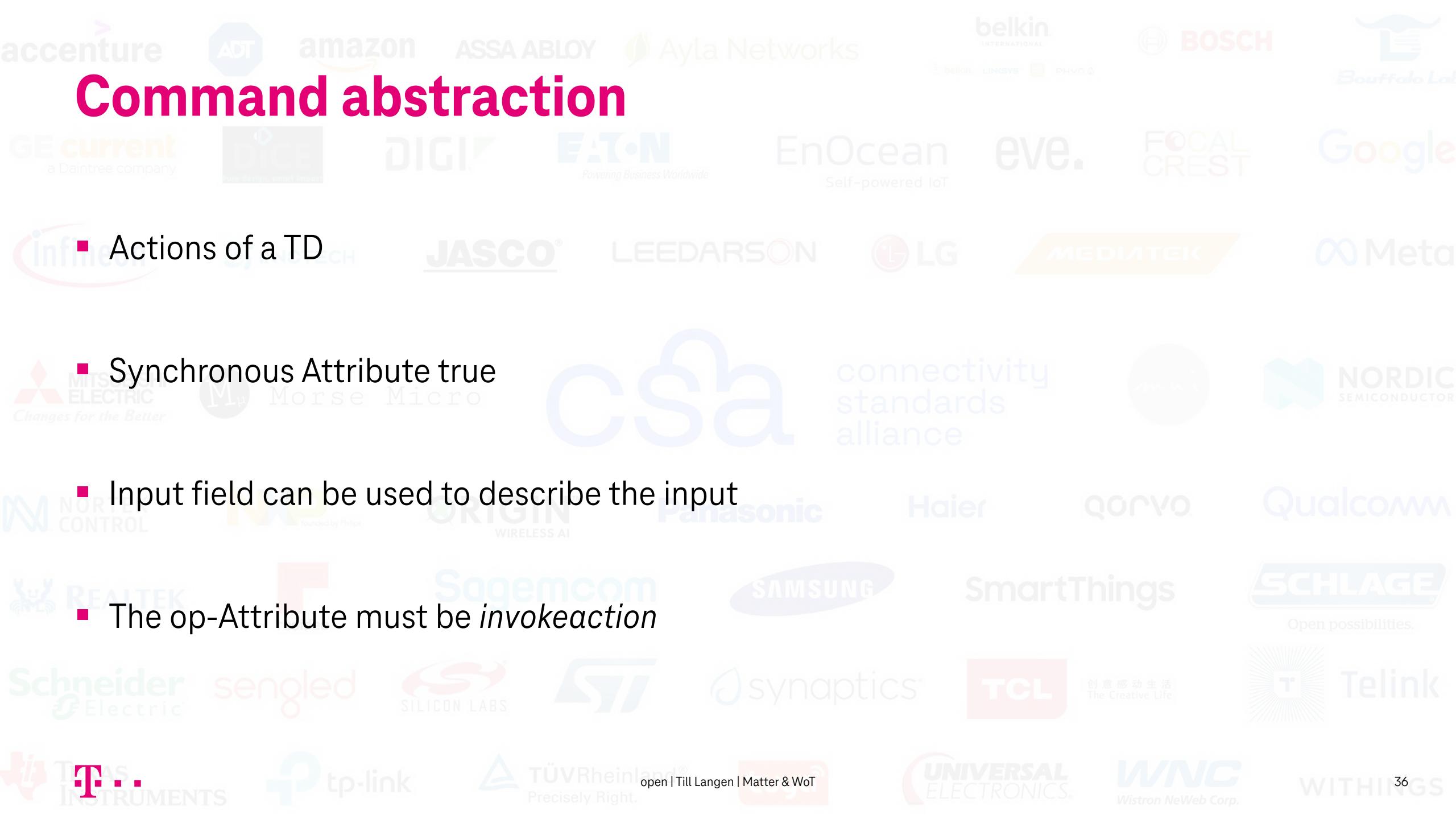
Attribute abstraction

- Properties of a TD
- The *readOnly* or *writeOnly* attribute for access control
- Observable attribute to false
- The op-Attribute *Form* must be *readproperty*

Attribute abstraction – an example

```
"OnOff:OnOff": {  
    "@type": "matter0x0006:0x0000",  
    "type": "boolean",  
    "readOnly": true,  
    "forms": [{  
        "href": "matter://1001/1/onoff",  
        "op": [  
            "readproperty"  
        ]},  
        "writeOnly": false,  
        "observable": false  
    }]
```



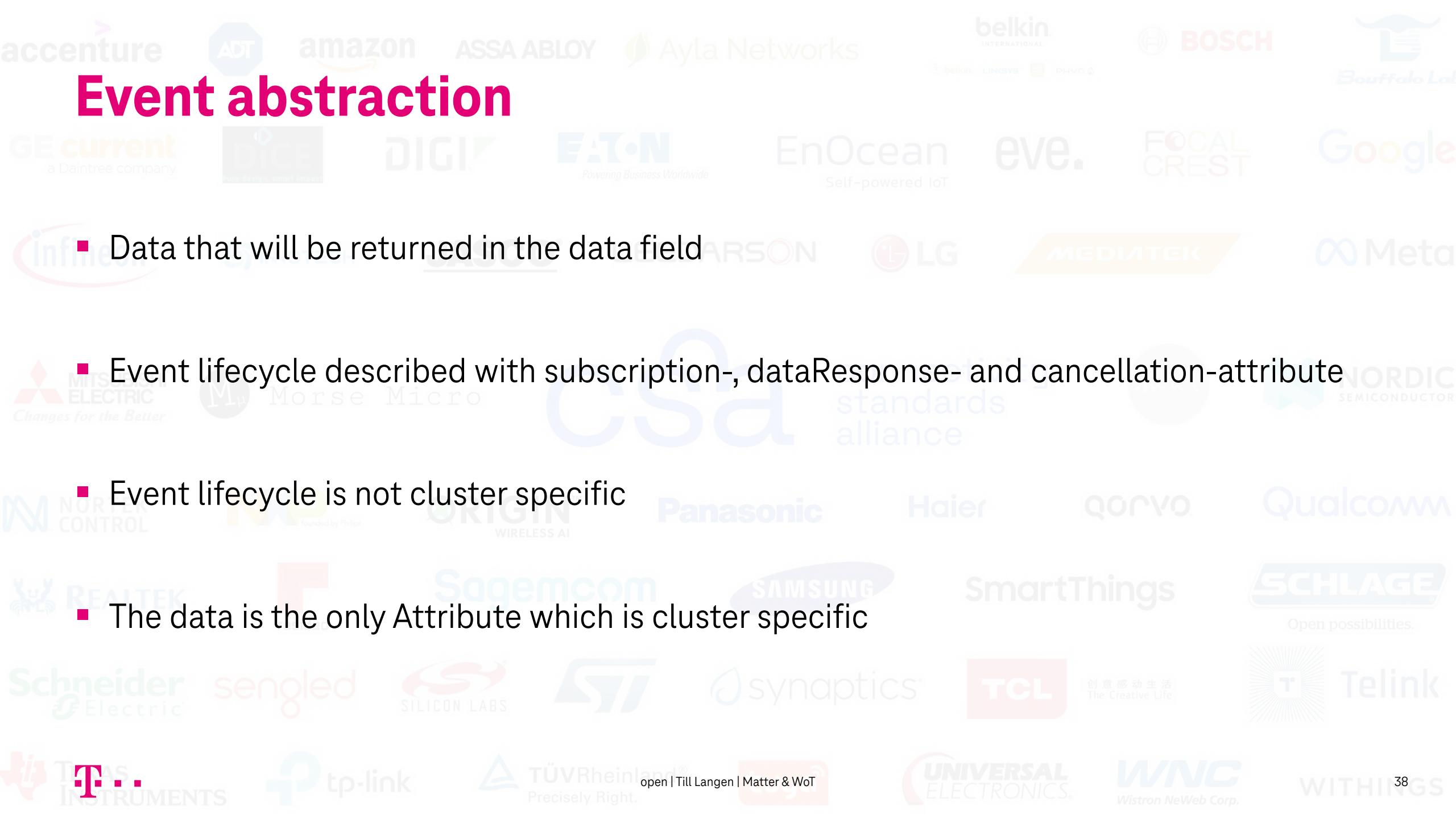


Command abstraction

- Actions of a TD
- Synchronous Attribute true
- Input field can be used to describe the input
- The op-Attribute must be *invokeaction*

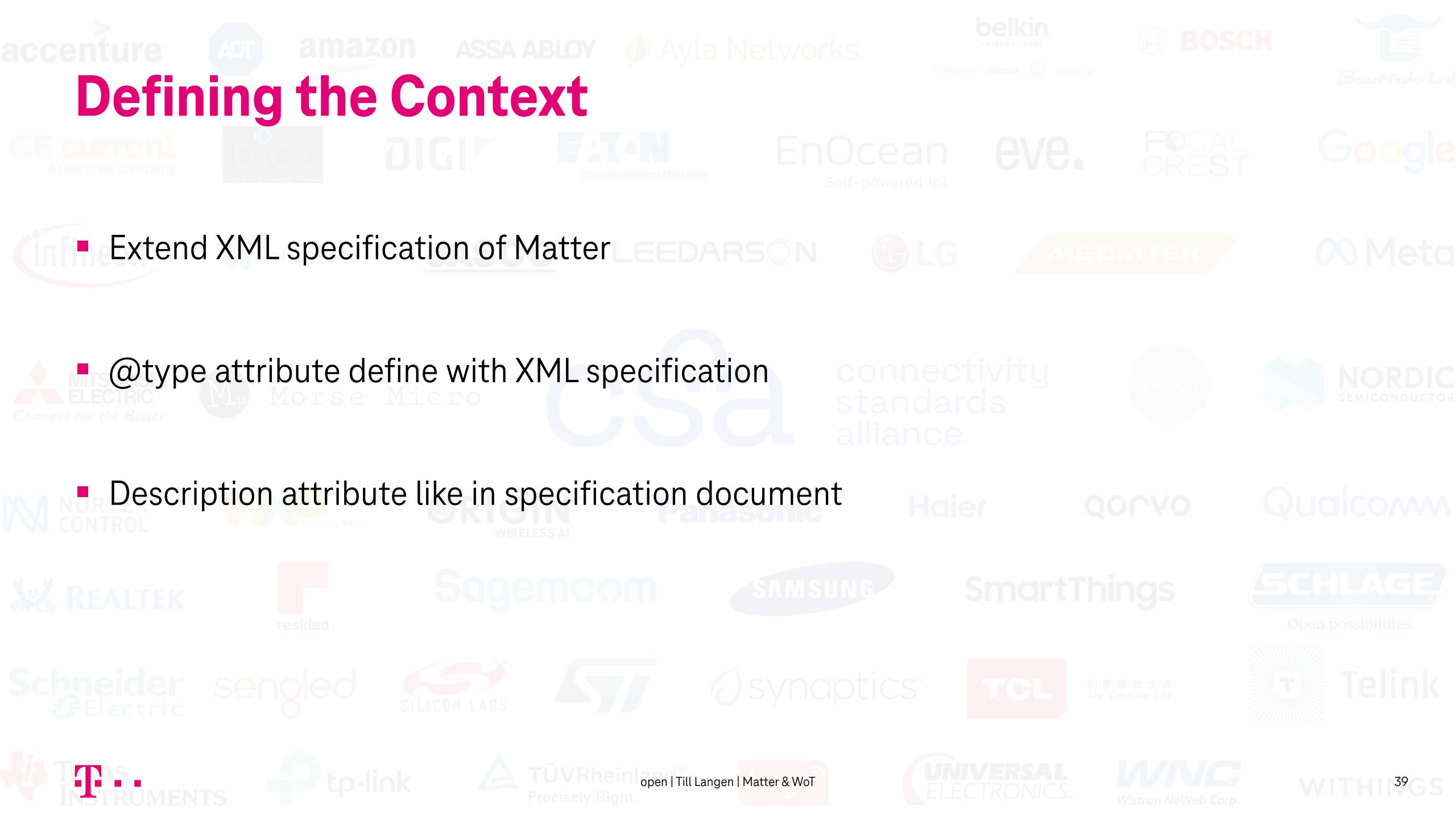
Command abstraction – an example

```
"OnOff:OffWithEffect": {  
    "@type": "matter0x0006:0x40",  
    "synchronous": true,  
    "input": {  
        "type": "object",  
        "properties": {...},  
        "required": [...]},  
    "forms": [{  
        "href": "matter://1001/1/onoff",  
        "op": ["invokeaction"]}],  
    "idempotent": true,  
}
```



Event abstraction

- Data that will be returned in the data field
- Event lifecycle described with subscription-, dataResponse- and cancellation-attribute
- Event lifecycle is not cluster specific
- The data is the only Attribute which is cluster specific



Defining the Context

- Extend XML specification of Matter
- @type attribute define with XML specification
- Description attribute like in specification document

Defining the Context – an example

{

"actions": {

 "OnOff:Toggle": {

 "@type": "matter0x0006:0x02",

 "description": "On receipt of the Toggle command, if the value of the OnOff attribute is equal to FALSE, the server SHALL set the OnOff attribute to TRUE, otherwise, the server SHALL set the OnOff attribute to FALSE."

}

}

}

T..
INSTRUMENTS

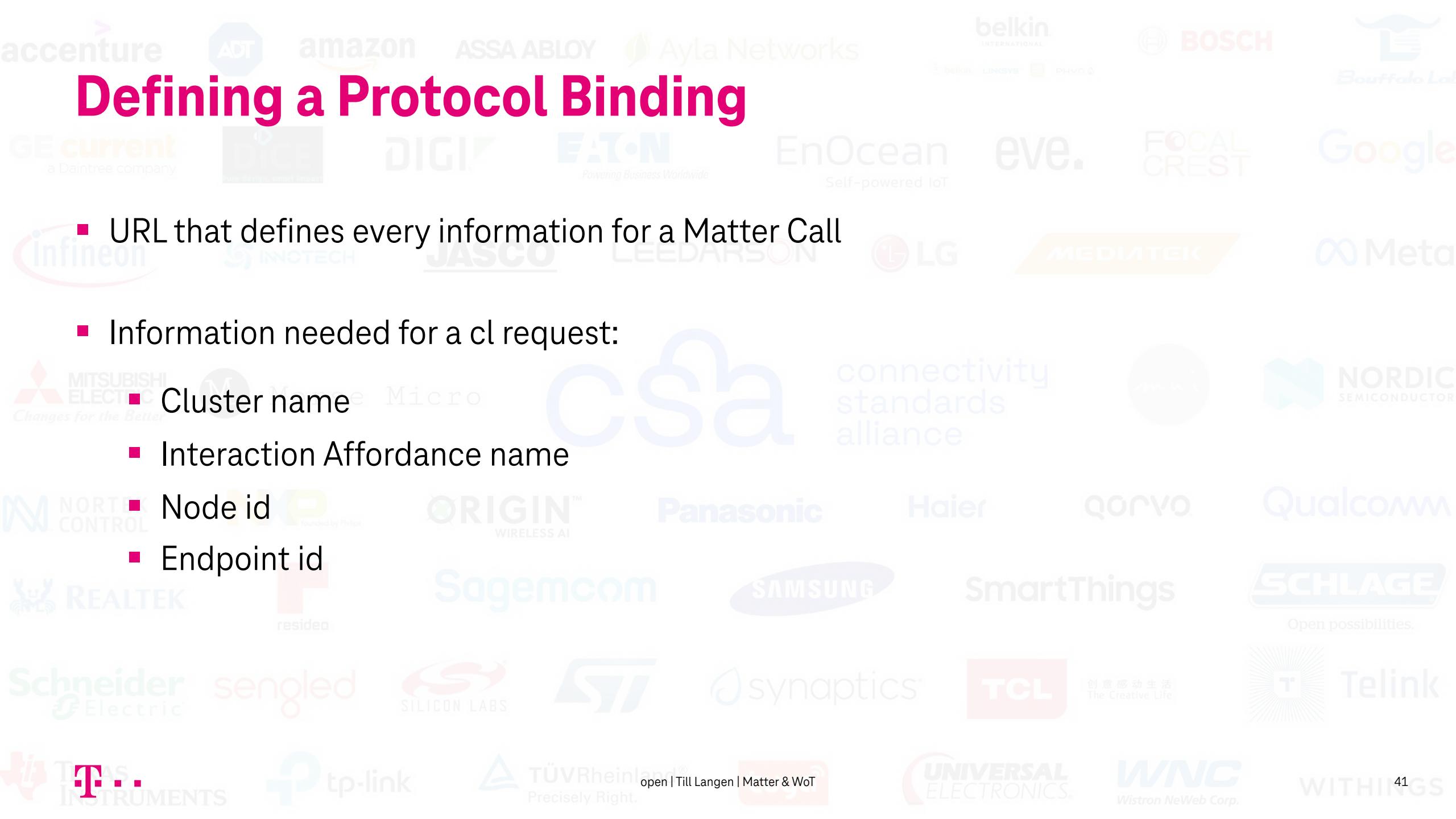
tp-link

TÜV Rheinland
open | Till Langen | Matter & WoT
Precisely Right.

UNIVERSAL
ELECTRONICS

VNC
Wistron NeWeb Corp.

WITHINGS



Defining a Protocol Binding

- URL that defines every information for a Matter Call

- Information needed for a cl request:

- Cluster name
 - Interaction Affordance name
- Node id
- Endpoint id

Defining a Protocol Binding

- Call read with Matter cl Tool:
\$./chip-tool <cluster_name> read <attribute_name> <node_id> <endpoint_id>
 - Translated to *href* attribute:
matter://<node_id>/<endpoint_id>/<cluster_name>/read/<attribute_name>

Defining a Protocol Binding

- Call **invoke** with Matter cl Tool:

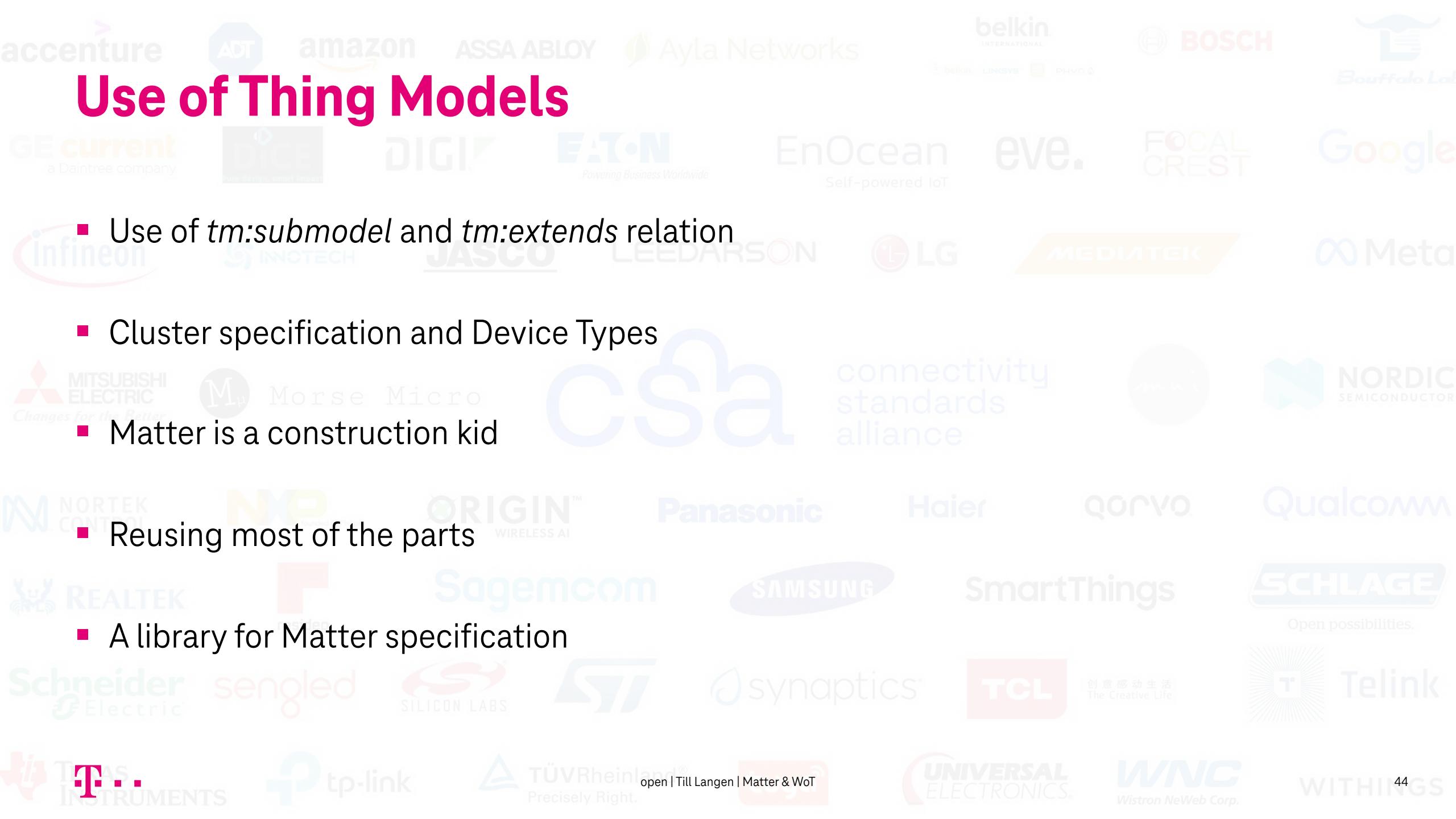
```
$ ./chip-tool <cluster_name><command_name><parameterX><node_id><endpoint_id>
```

- Translated to *href* attribute:

```
matter://<node_id>/<endpoint_id>/<cluster_name>/invoke/<command_name>?<parameterX>=<value>
```

- Same for the **subscription** of an Event:

```
matter://<node_id>/<endpoint_id>/<cluster_name>/subscribe/<event_name>?<parameterX>=<value>
```



Use of Thing Models

- Use of *tm:submodel* and *tm:extends* relation

- Cluster specification and Device Types

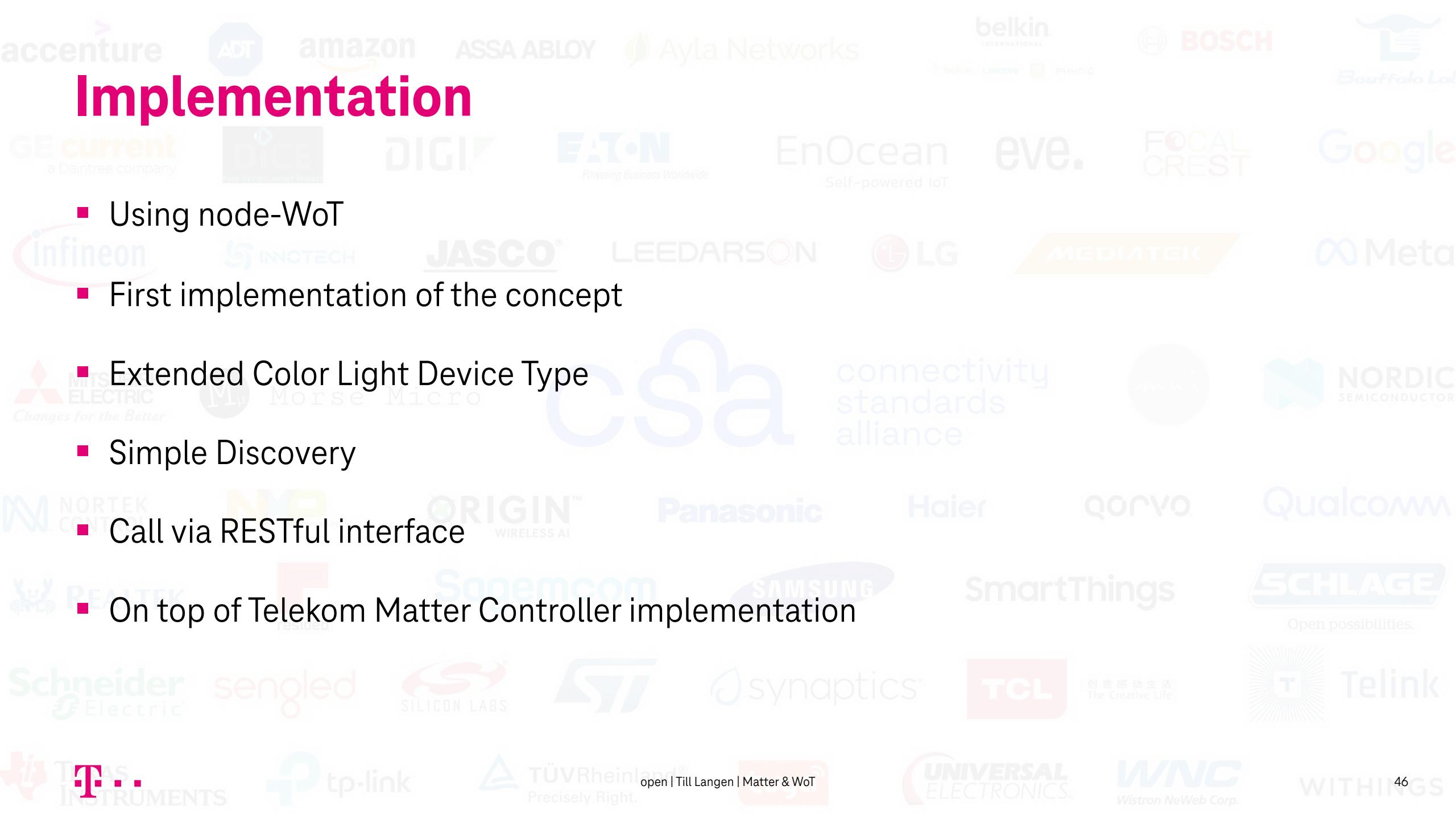
- Matter is a construction kid

- Reusing most of the parts

- A library for Matter specification

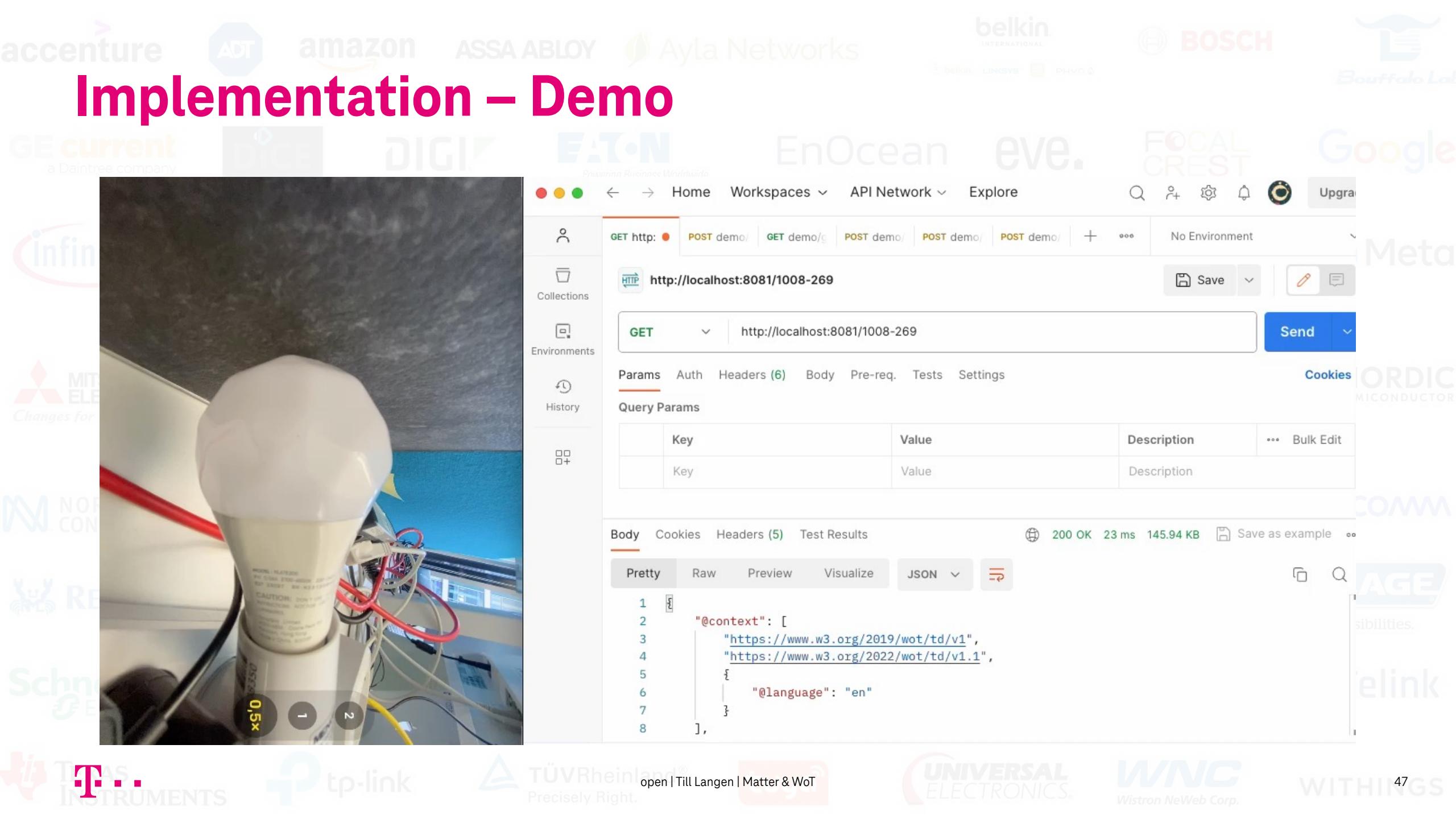


Implementation



Implementation

- Using node-WoT
- First implementation of the concept
- Extended Color Light Device Type
- Simple Discovery
- Call via RESTful interface
- On top of Telekom Matter Controller implementation



Implementation – Demo



EATON Powering Business Worldwide

Home Workspaces API Network Explore

GET http://localhost:8081/1008-269

Send

Params Auth Headers (6) Body Pre-req. Tests Settings Cookies

Query Params

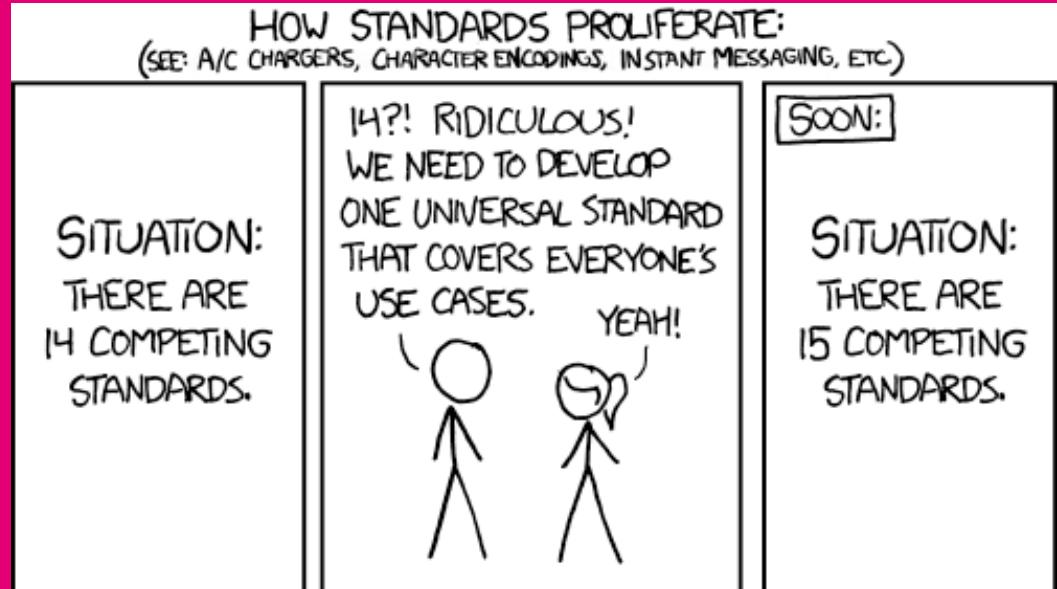
Key	Value	Description
Key	Value	Description

Body Cookies Headers (5) Test Results

Pretty Raw Preview Visualize JSON

```
1  "@context": [
2    "https://www.w3.org/2019/wot/td/v1",
3    "https://www.w3.org/2022/wot/td/v1.1",
4    {
5      "@language": "en"
6    }
7  ],
```

Time for Questions





Backup



Repository Link

- [Here](#)