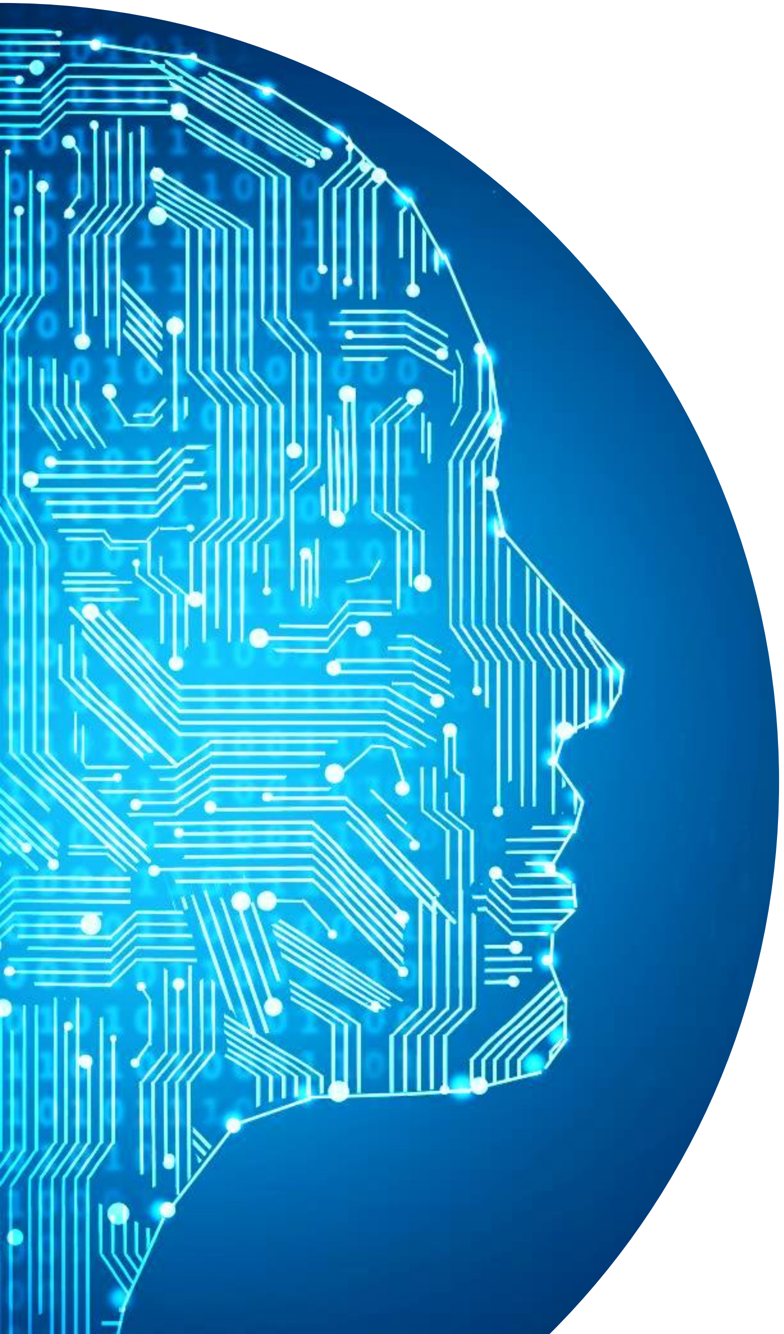
TeraFlow SDN

by ETSI

Controller Abstractions

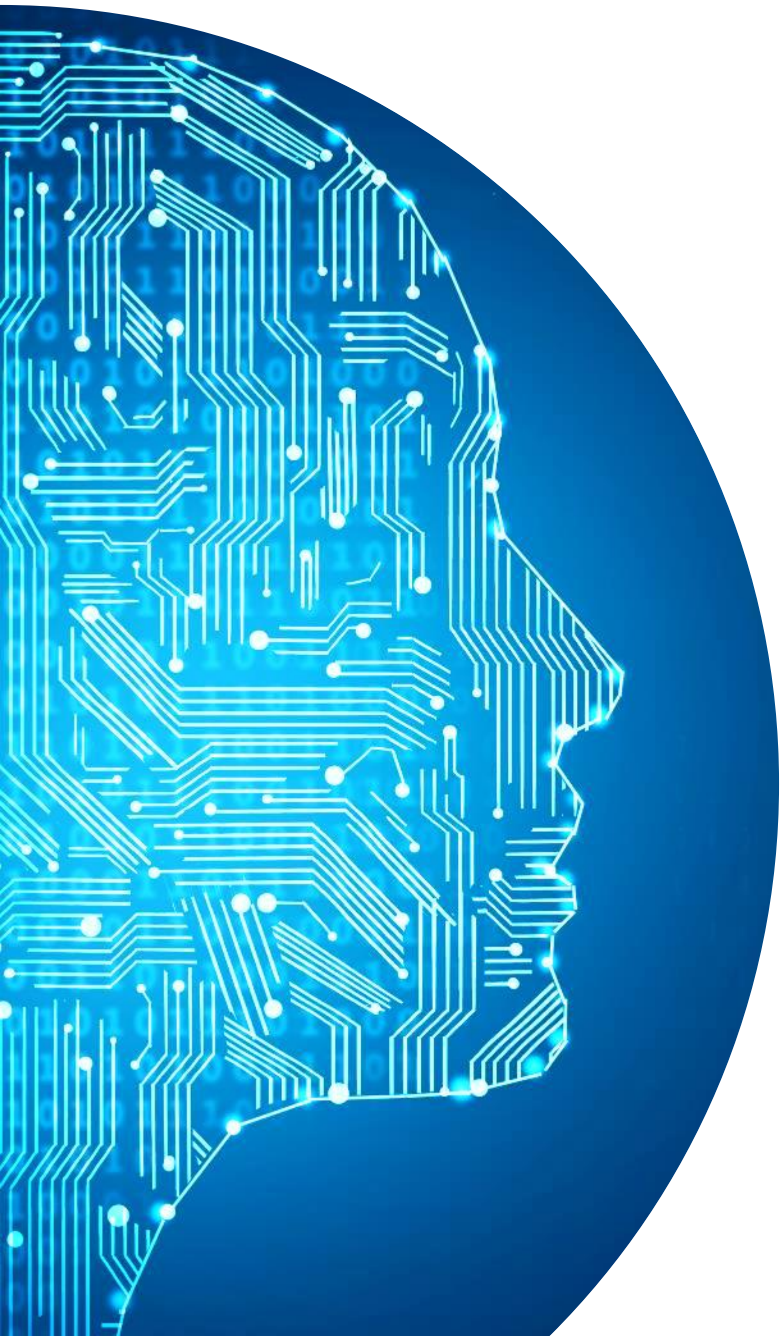
Georgios P. Katsikas, Panagiotis Famelis

ETSI TFS – Hackfest #3, October 16, 2023

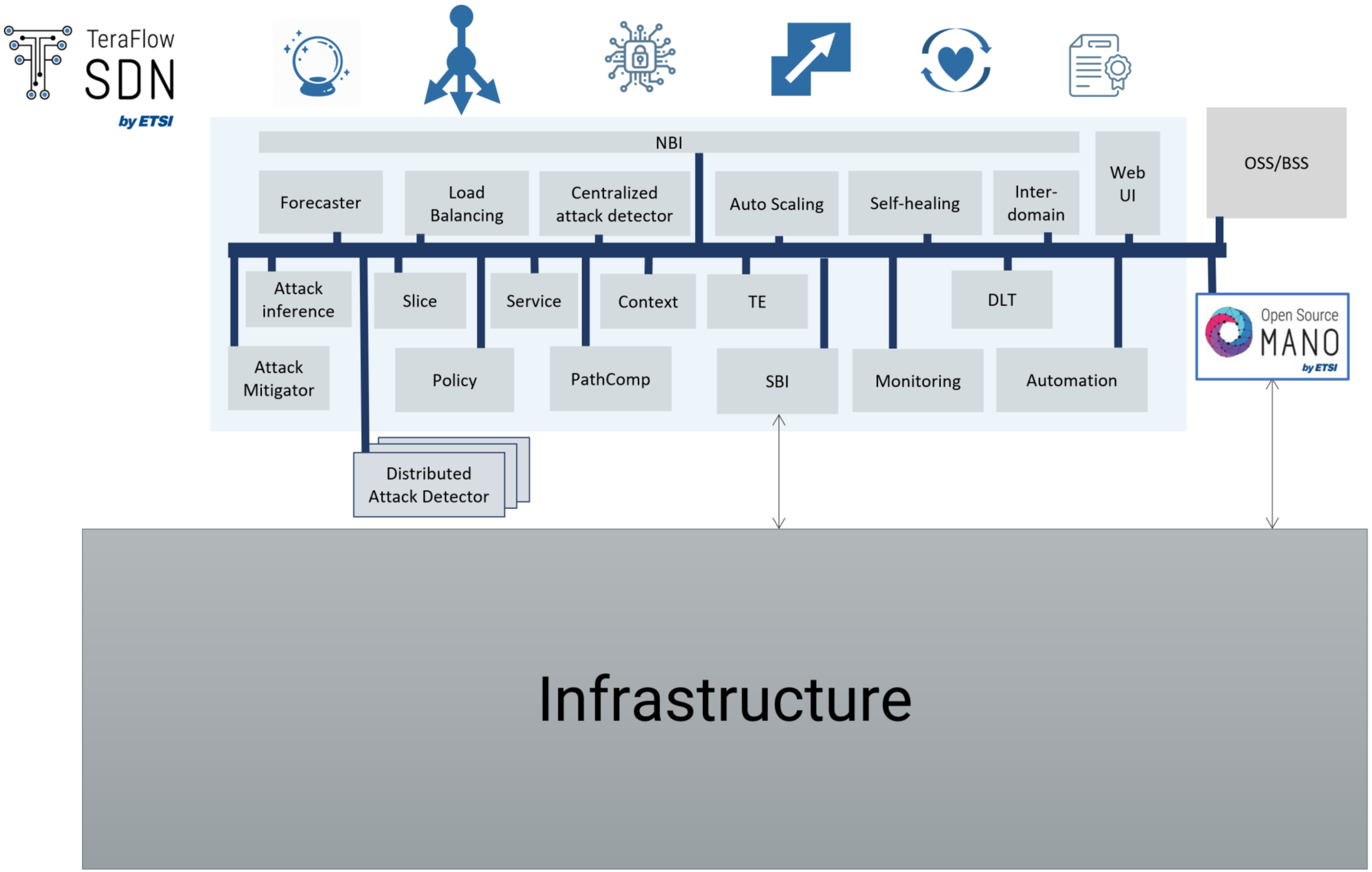


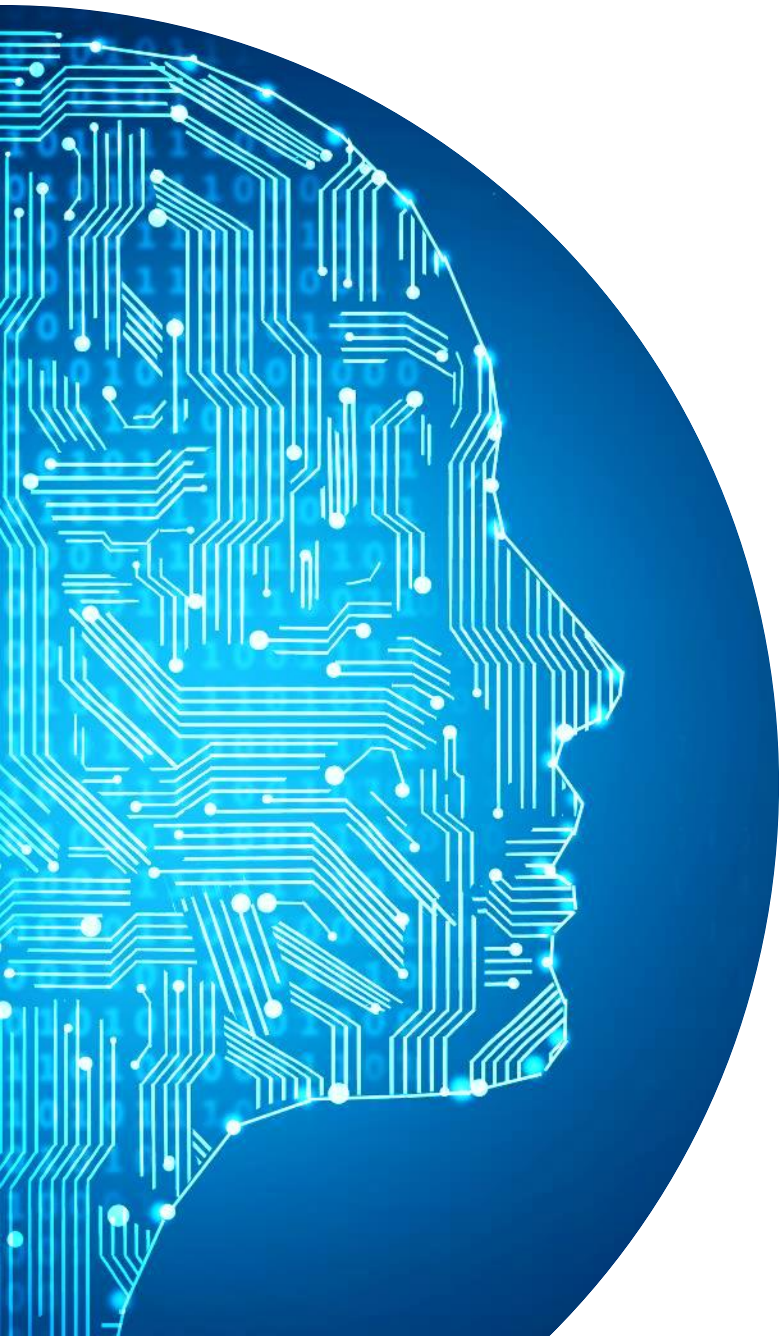
Agenda

- ETSI TFS Architecture
- Abstractions
  - Device-level
  - Service-level
  - Management-level
- Summary

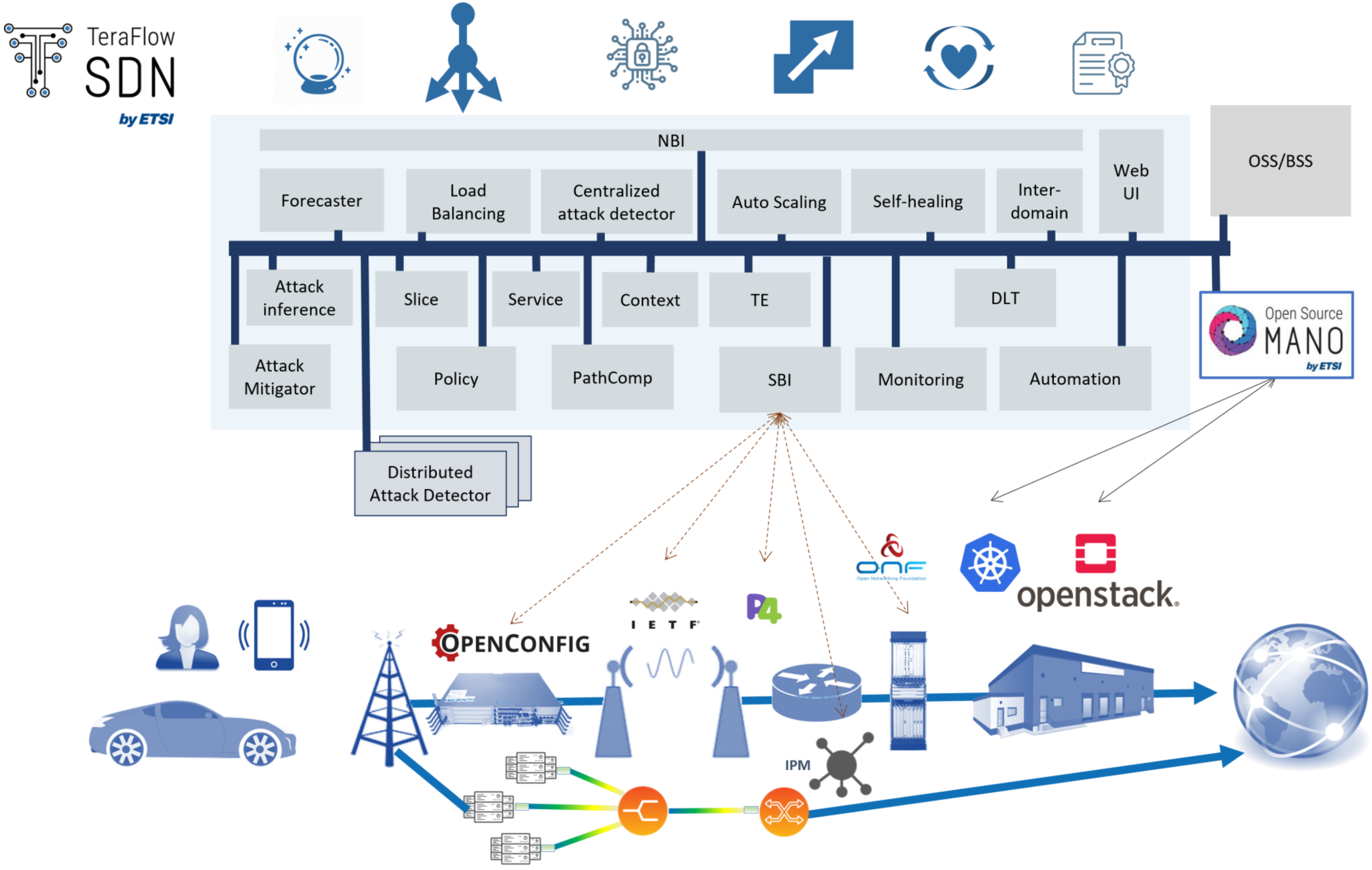


ETSI TFS Architecture





ETSI TFS Architecture



NBI

Forecaster

Load Balancing

Centralized attack detector

Auto Scaling

Self-healing

Inter-domain

Web UI

OSS/BSS

Attack inference

Slice

Service

Context

TE

DLT

Attack Mitigator

Policy

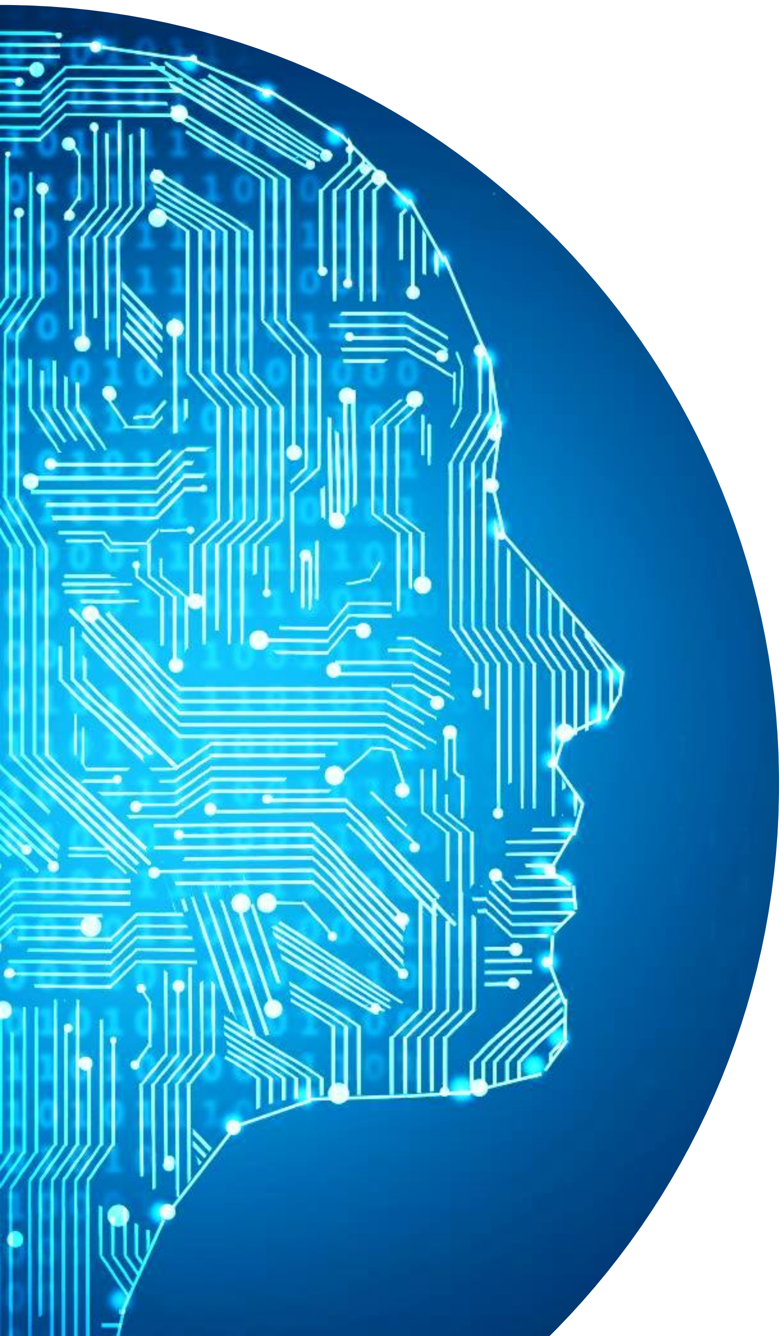
PathComp

SBI

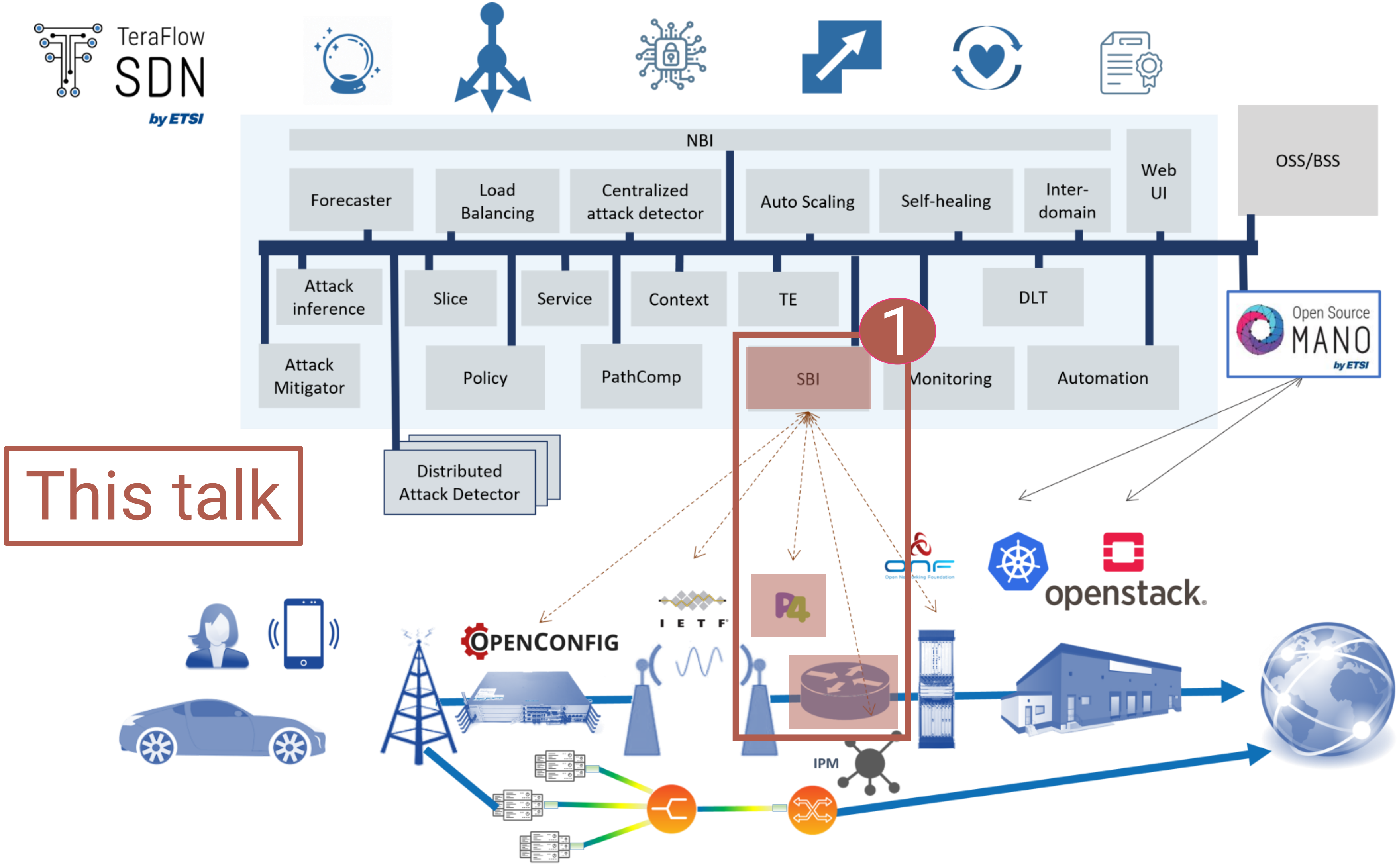
Monitoring

Automation

Distributed Attack Detector



ETSI TFS Architecture



TeraFlow SDN by ETSI

NBI

Forecaster

Load Balancing

Centralized attack detector

Auto Scaling

Self-healing

Inter-domain

Web UI

OSS/BSS

Attack inference

Slice

Service

Context

TE

DLT

Open Source MANO by ETSI

Attack Mitigator

Policy

PathComp

SBI

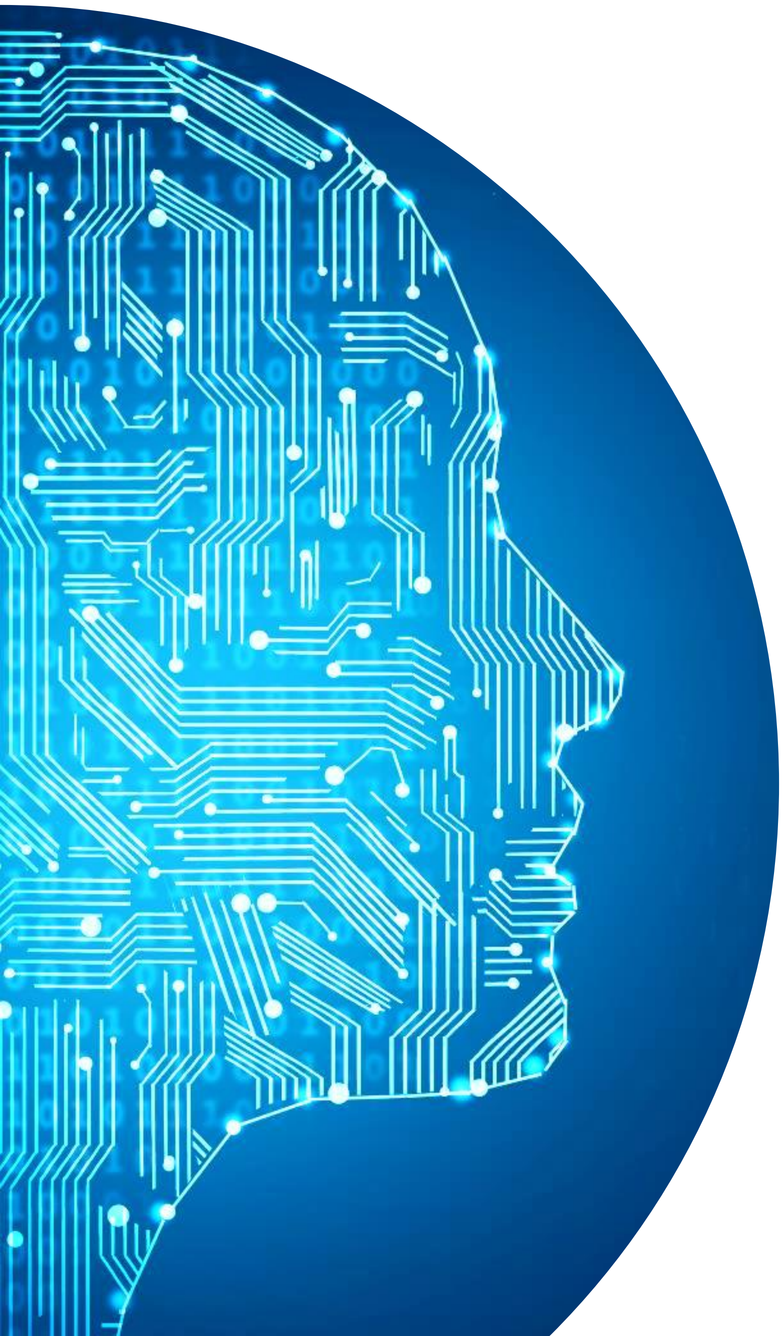
Monitoring

Automation

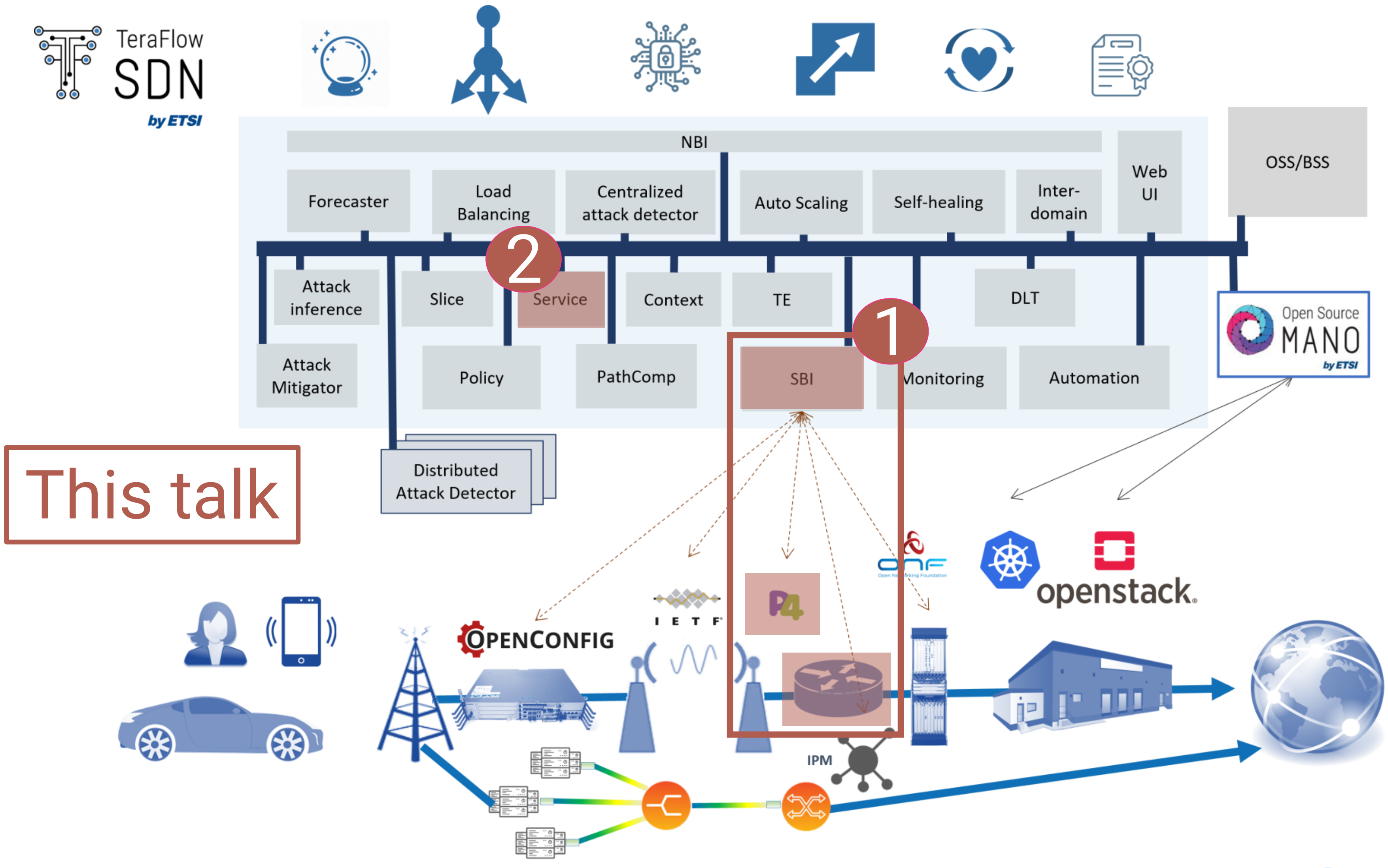
1

This talk

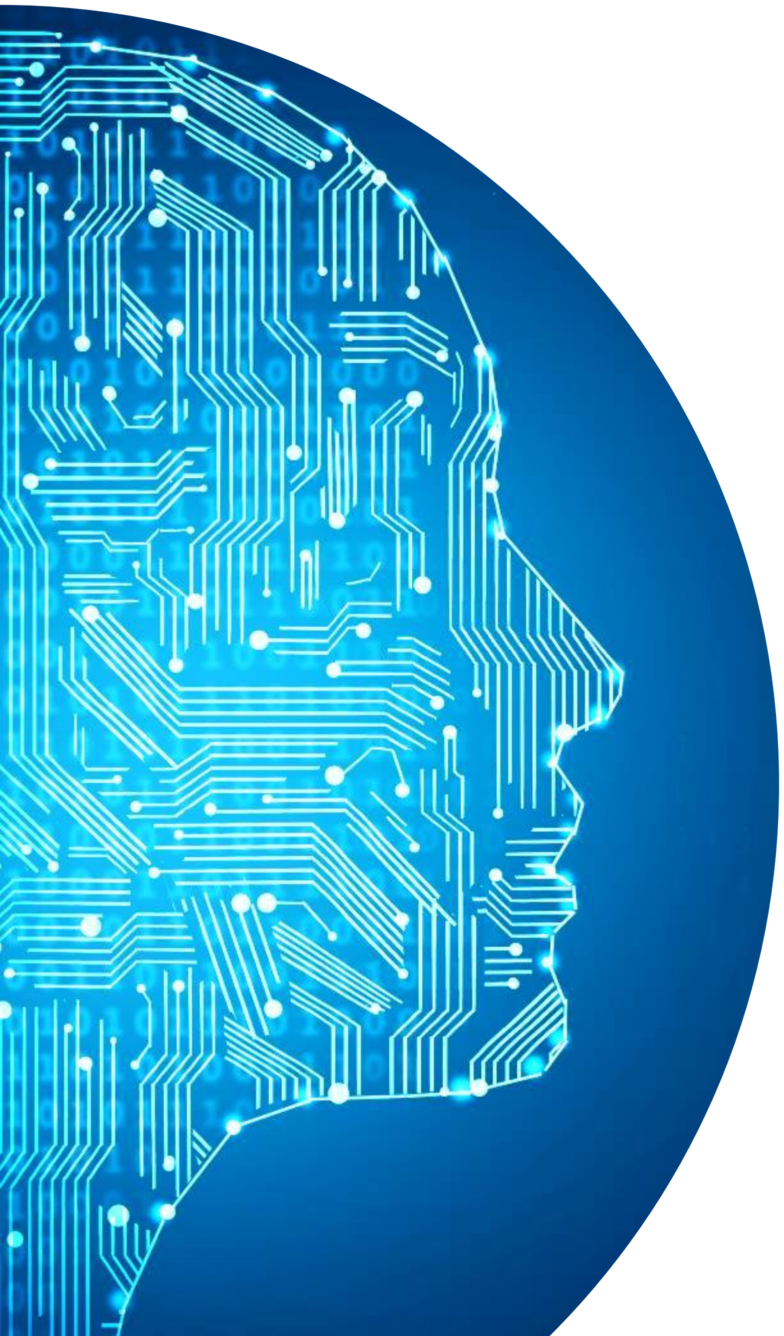
Distributed Attack Detector

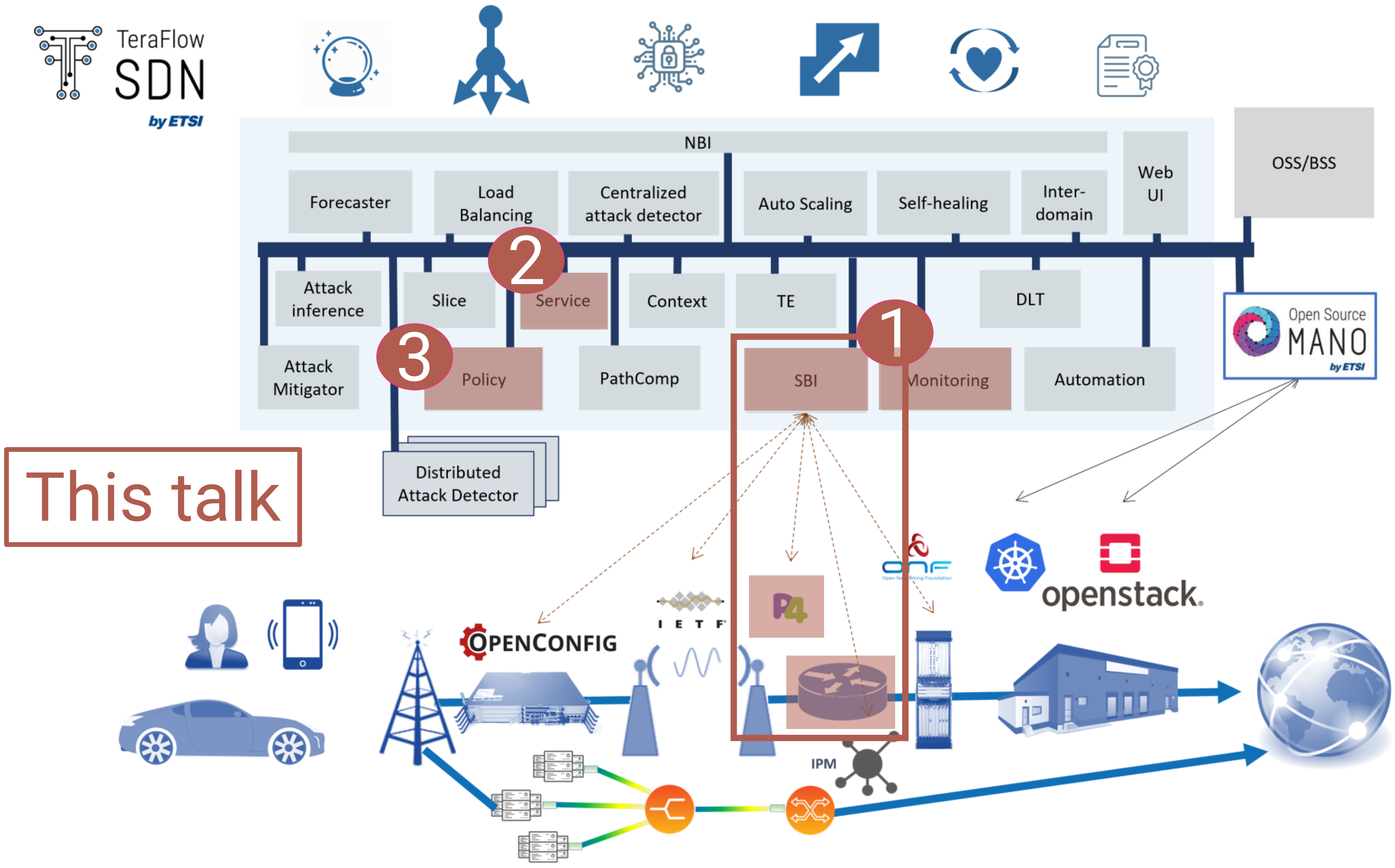


ETSI TFS Architecture



ETSI TFS Architecture





TeraFlow SDN by ETSI

NBI

Forecaster

Load Balancing

Centralized attack detector

Auto Scaling

Self-healing

Inter-domain

Web UI

OSS/BSS

Attack inference

Slice

Service

Context

TE

DLT

Attack Mitigator

Policy

PathComp

SBI

Monitoring

Automation

Open Source MANO by ETSI

Distributed Attack Detector

This talk

1

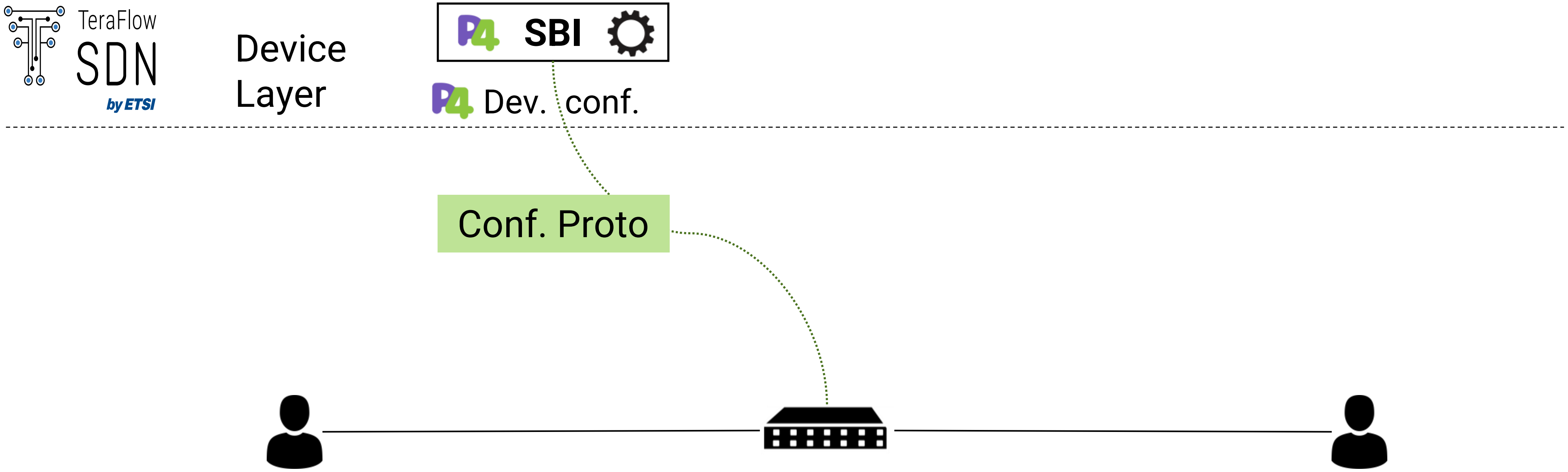
2

3

© ETSI CC-BY-4.0

Device-level Abstractions

ETSI TFS SBI microservice



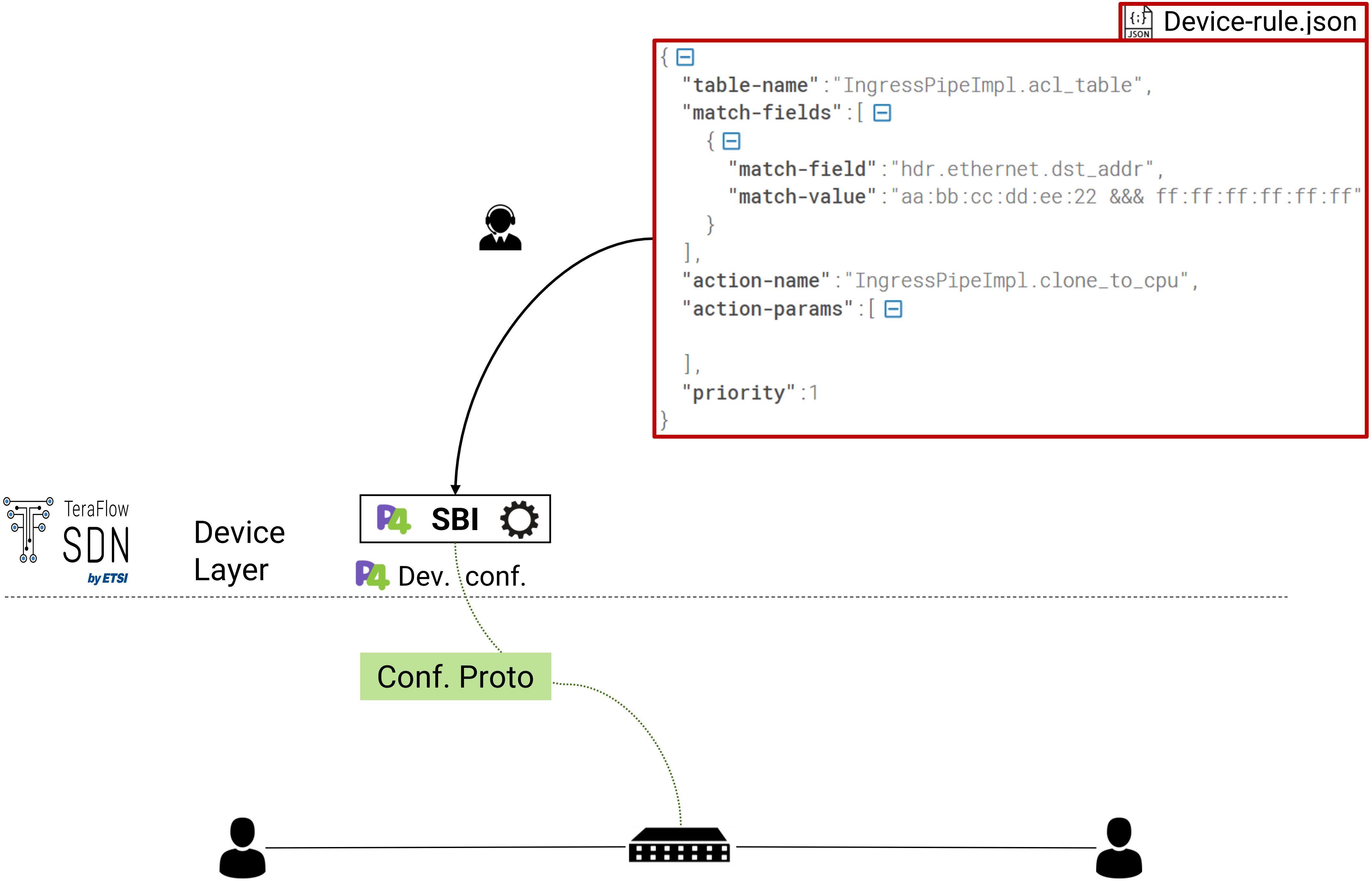
TeraFlow SDN by ETSI

Device Layer

SBI

Dev. conf.

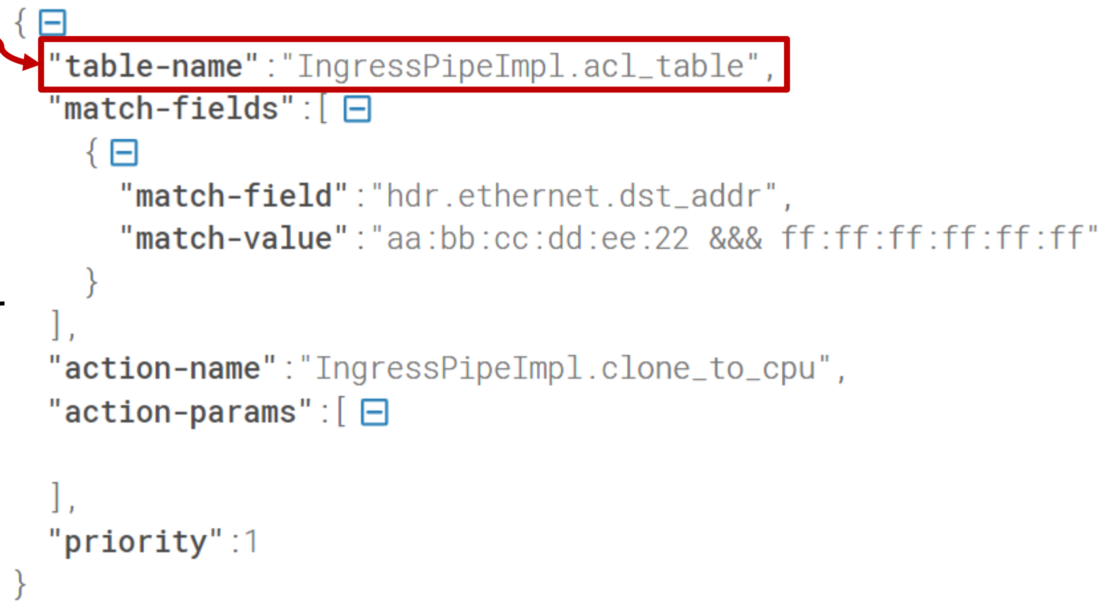
Conf. Proto



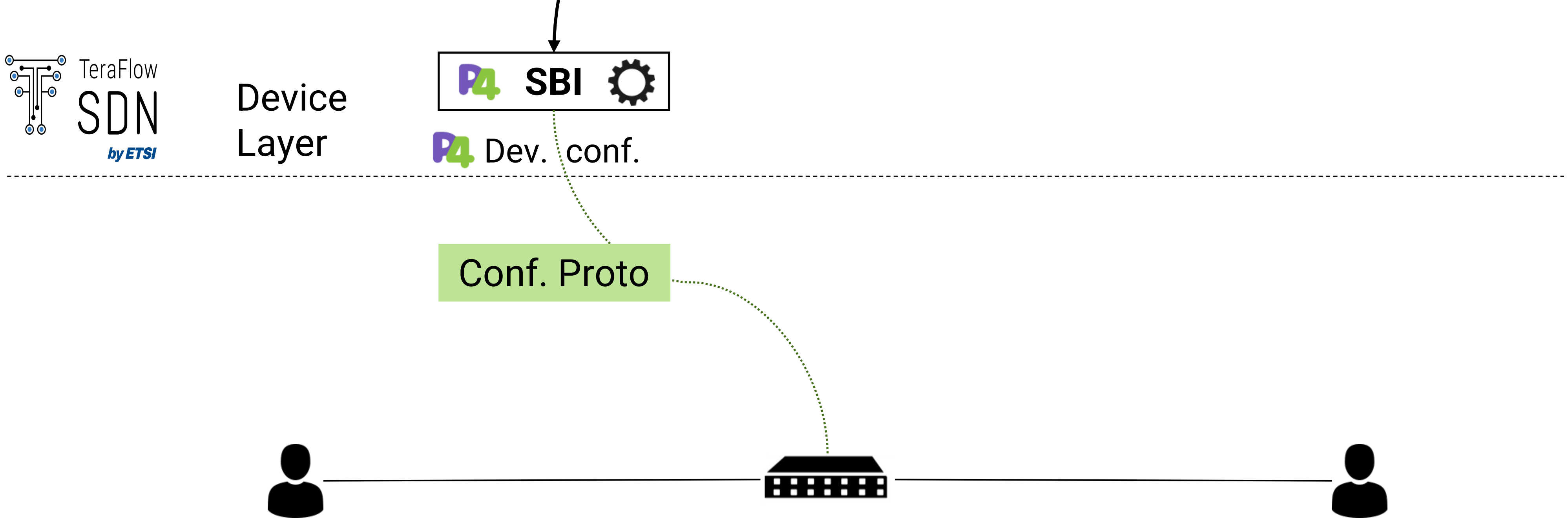
Device-rule.json

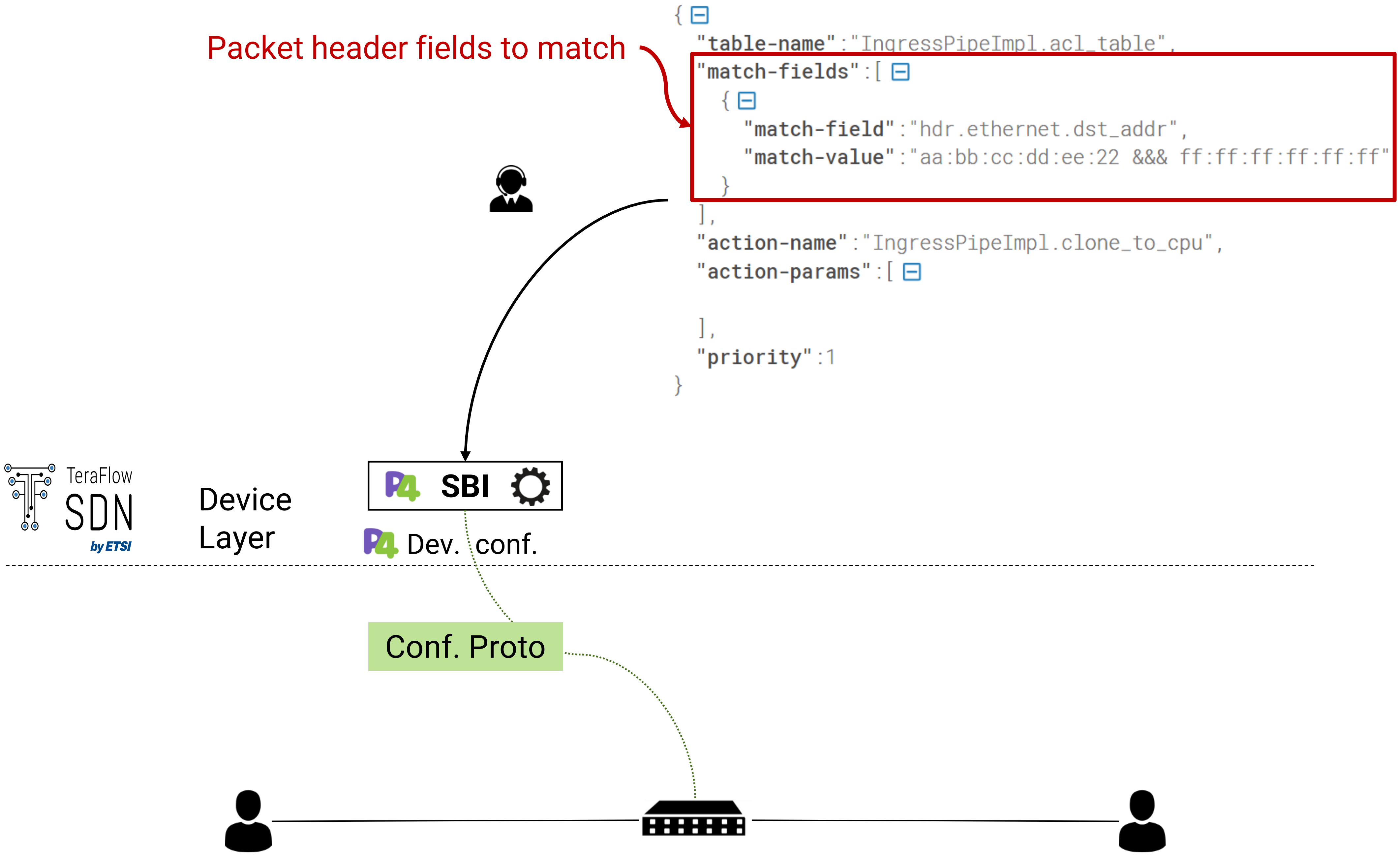
```
{ ⊟
  "table-name":"IngressPipeImpl.acl_table",
  "match-fields":[ ⊟
    { ⊟
      "match-field":"hdr.ethernet.dst_addr",
      "match-value":"aa:bb:cc:dd:ee:22 &&& ff:ff:ff:ff:ff:ff"
    }
  ],
  "action-name":"IngressPipeImpl.clone_to_cpu",
  "action-params":[ ⊟

  ],
  "priority":1
}
```

Pipeline table name, where the rule will be inserted



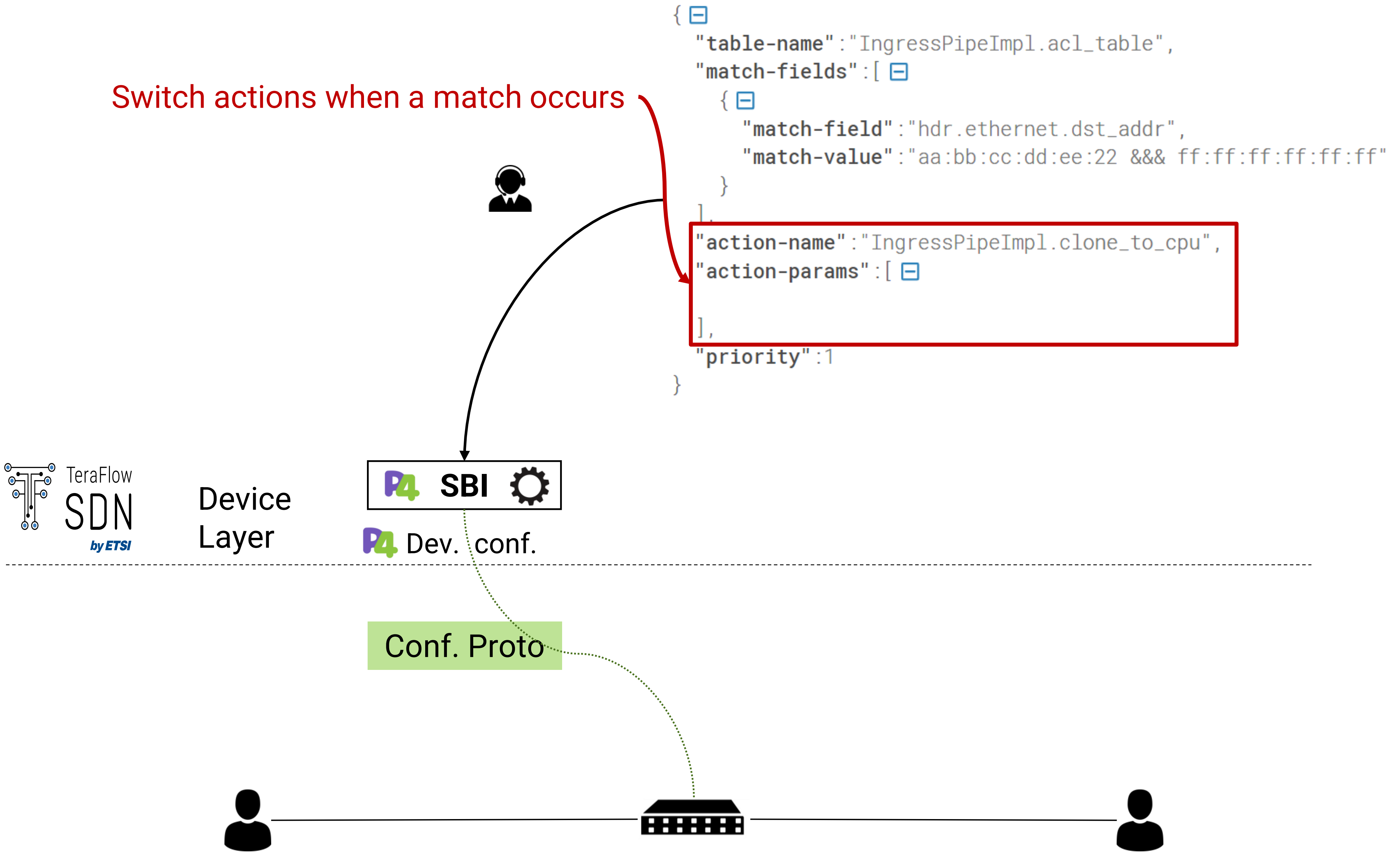
```
{
  "table-name":"IngressPipeImpl.acl_table",
  "match-fields":[
    {
      "match-field":"hdr.ethernet.dst_addr",
      "match-value":"aa:bb:cc:dd:ee:22 &&& ff:ff:ff:ff:ff:ff"
    }
  ],
  "action-name":"IngressPipeImpl.clone_to_cpu",
  "action-params":[

  ],
  "priority":1
}
```





Packet header fields to match

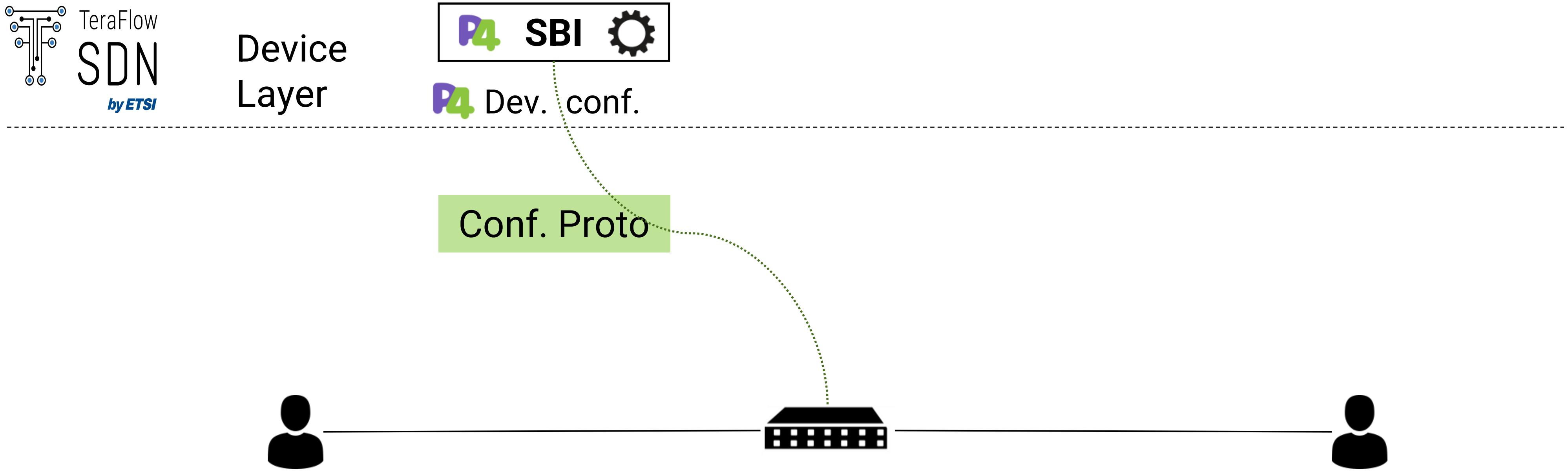
```
{
  "table-name":"IngressPipeImpl.acl_table",
  "match-fields":[
    {
      "match-field":"hdr.ethernet.dst_addr",
      "match-value":"aa:bb:cc:dd:ee:22 &&& ff:ff:ff:ff:ff:ff"
    }
  ],
  "action-name":"IngressPipeImpl.clone_to_cpu",
  "action-params":[

  ],
  "priority":1
}
```



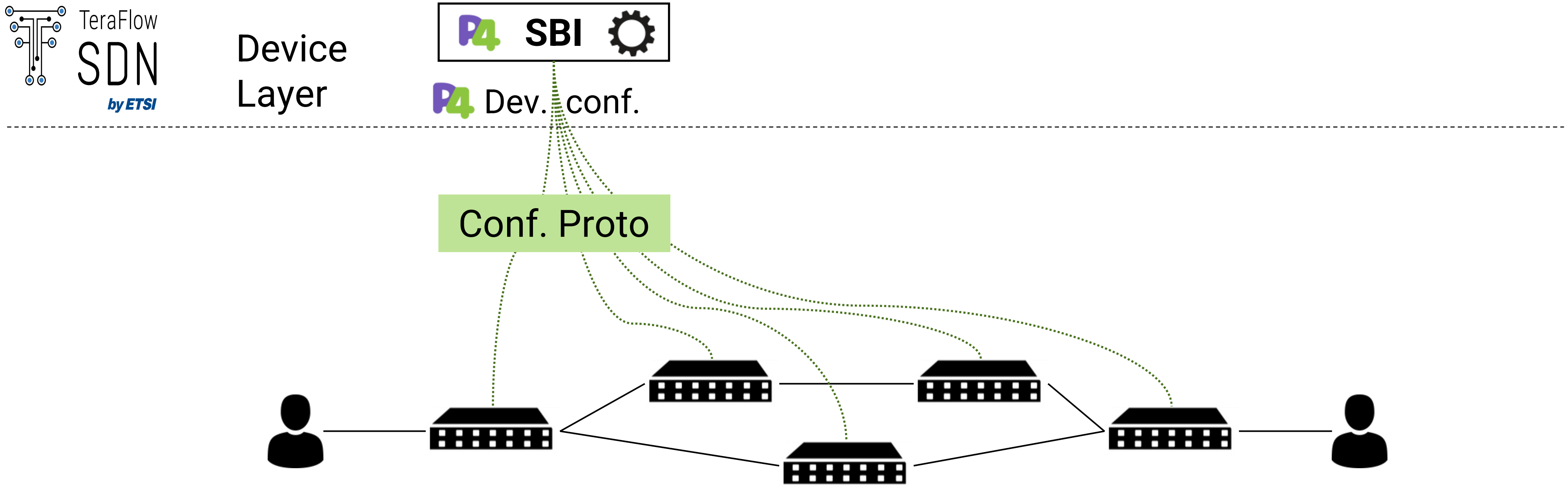
Switch actions when a match occurs

```
{ ⊟
  "table-name":"IngressPipeImpl.acl_table",
  "match-fields":[ ⊟
    { ⊟
      "match-field":"hdr.ethernet.dst_addr",
      "match-value":"aa:bb:cc:dd:ee:22 &&& ff:ff:ff:ff:ff:ff"
    }
  ],
  "action-name":"IngressPipeImpl.clone_to_cpu",
  "action-params":[ ⊟

  ],
  "priority":1
}
```

Managing a single device is not a problem

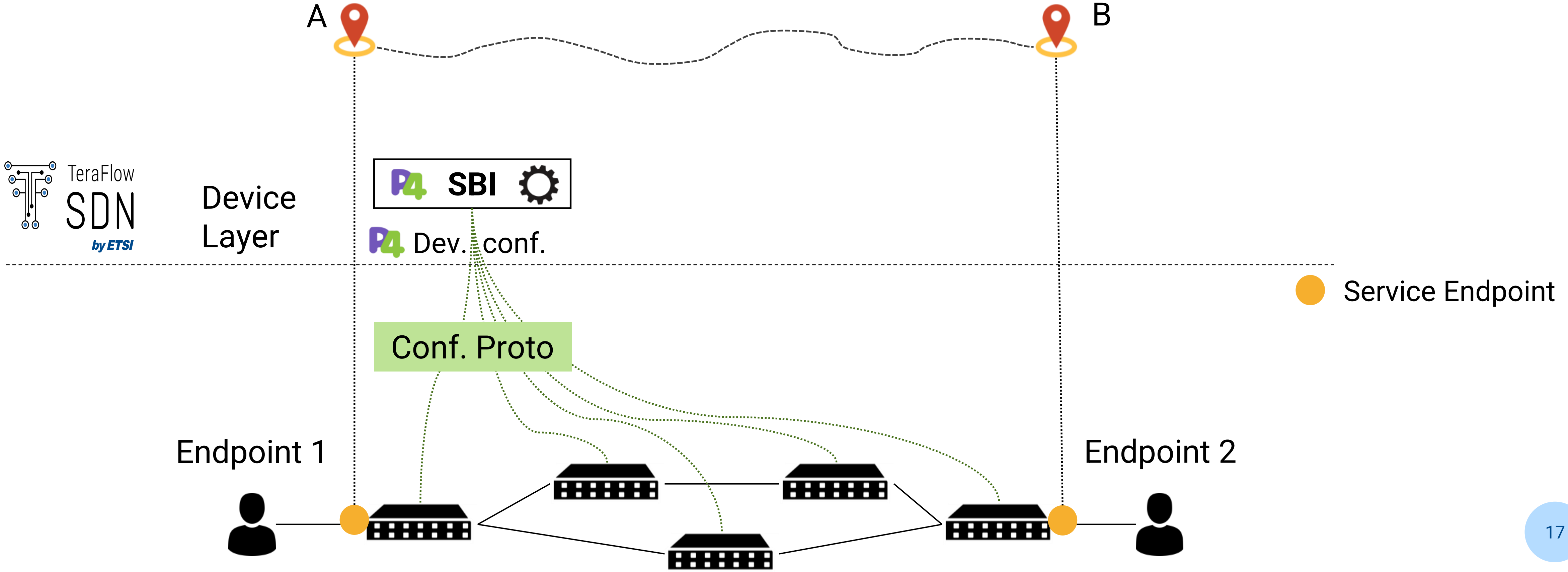


What about larger topologies, or worse, real networks?



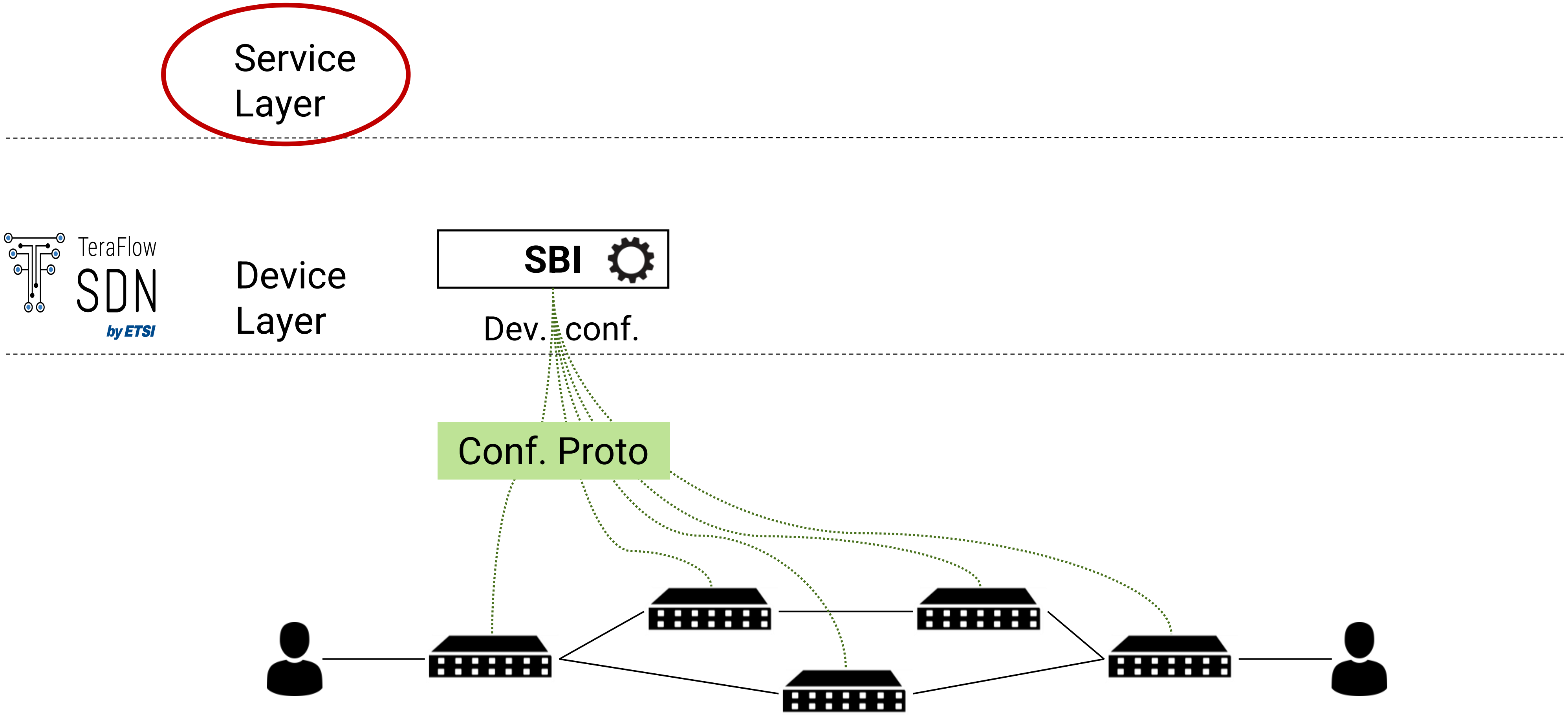
What about larger topologies, or worse, real networks?

Necessary abstraction: Connectivity intent from A to B



Service-level Abstractions

ETSI TFS Service microservice



Service Layer

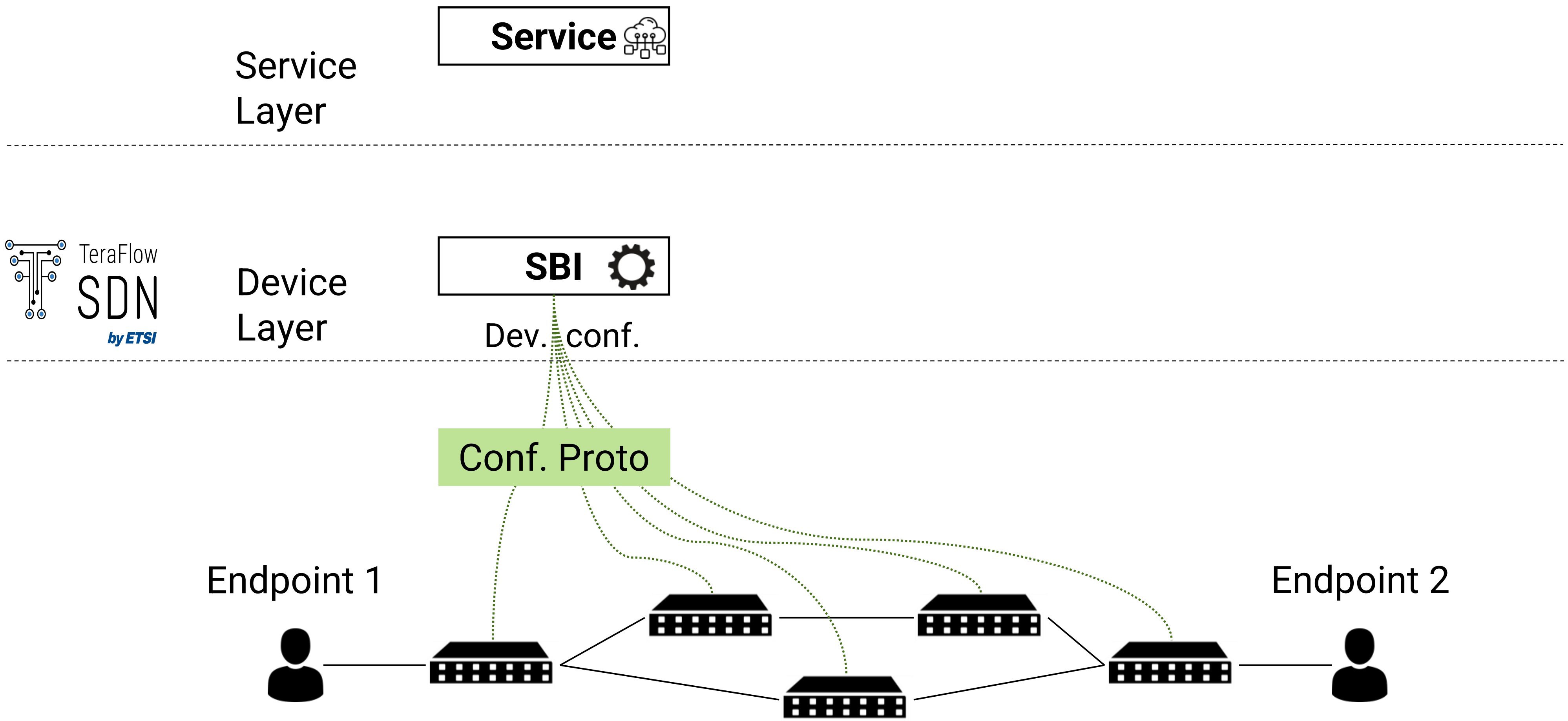
Device Layer

SBI

Dev. conf.

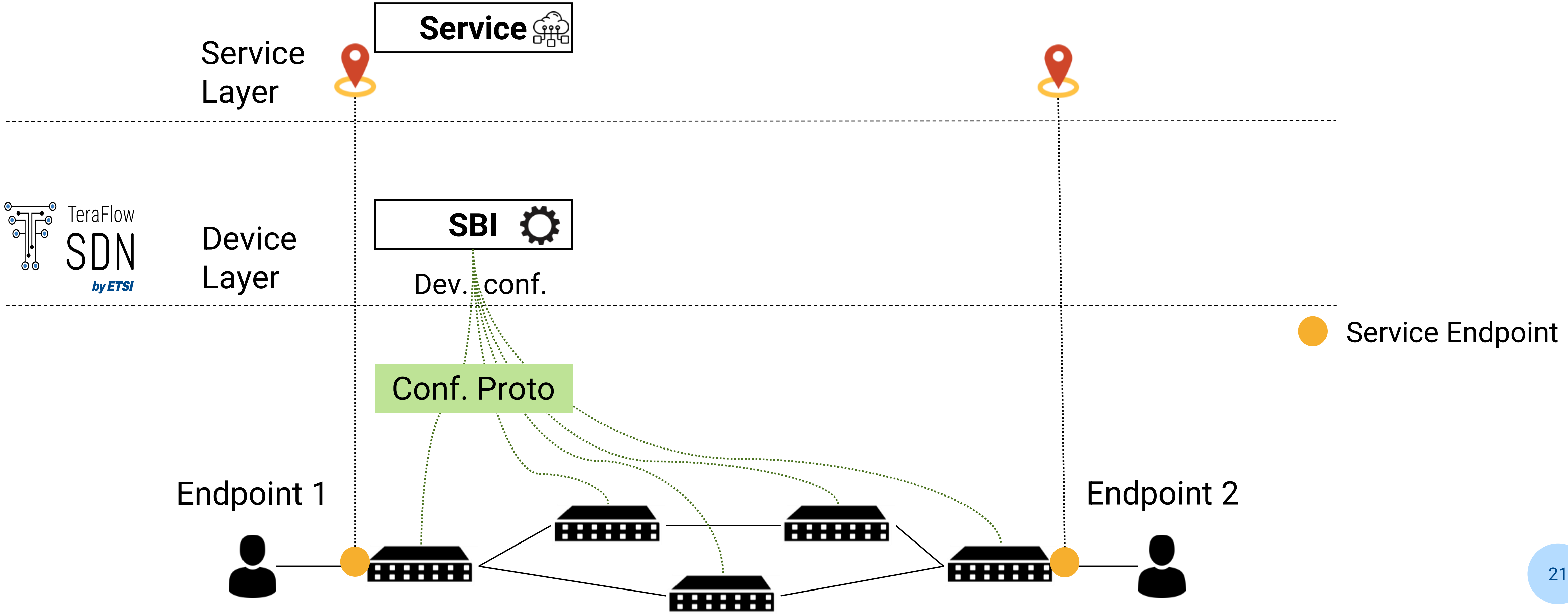
Conf. Proto

Service abstractions



Service abstractions

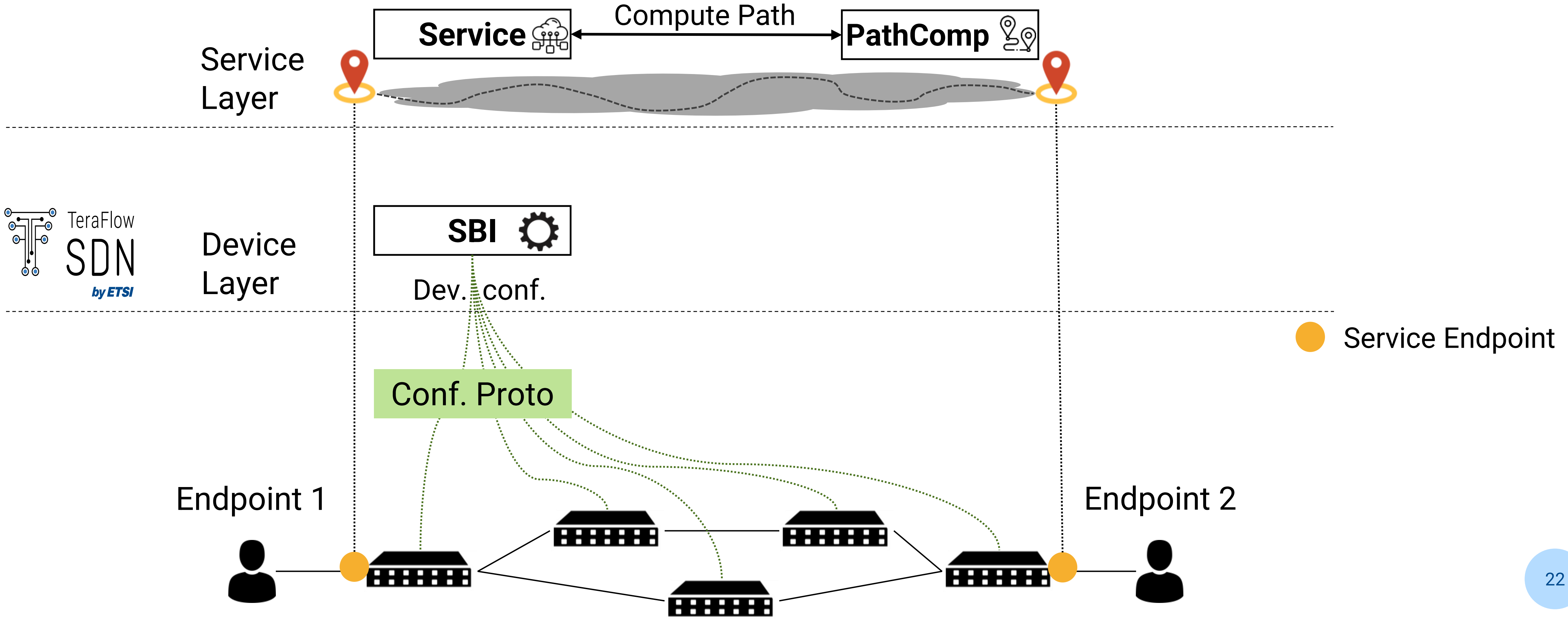
End-to-end connectivity intents between endpoints



Service abstractions

End-to-end connectivity intents between endpoints

Modular path computation between endpoints

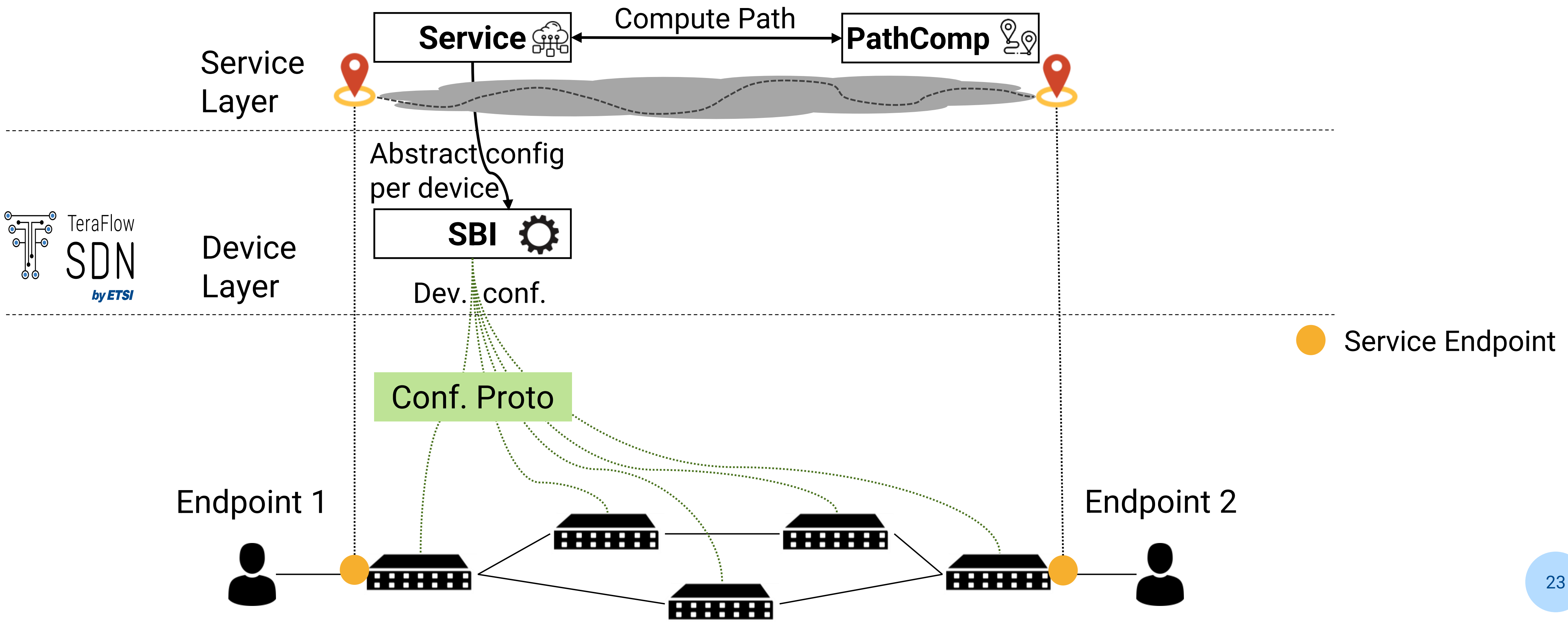


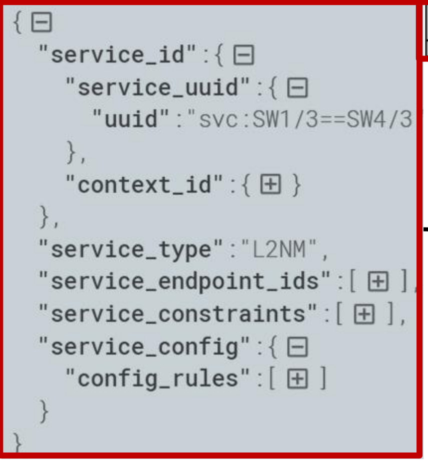
Service abstractions

End-to-end connectivity intents between endpoints

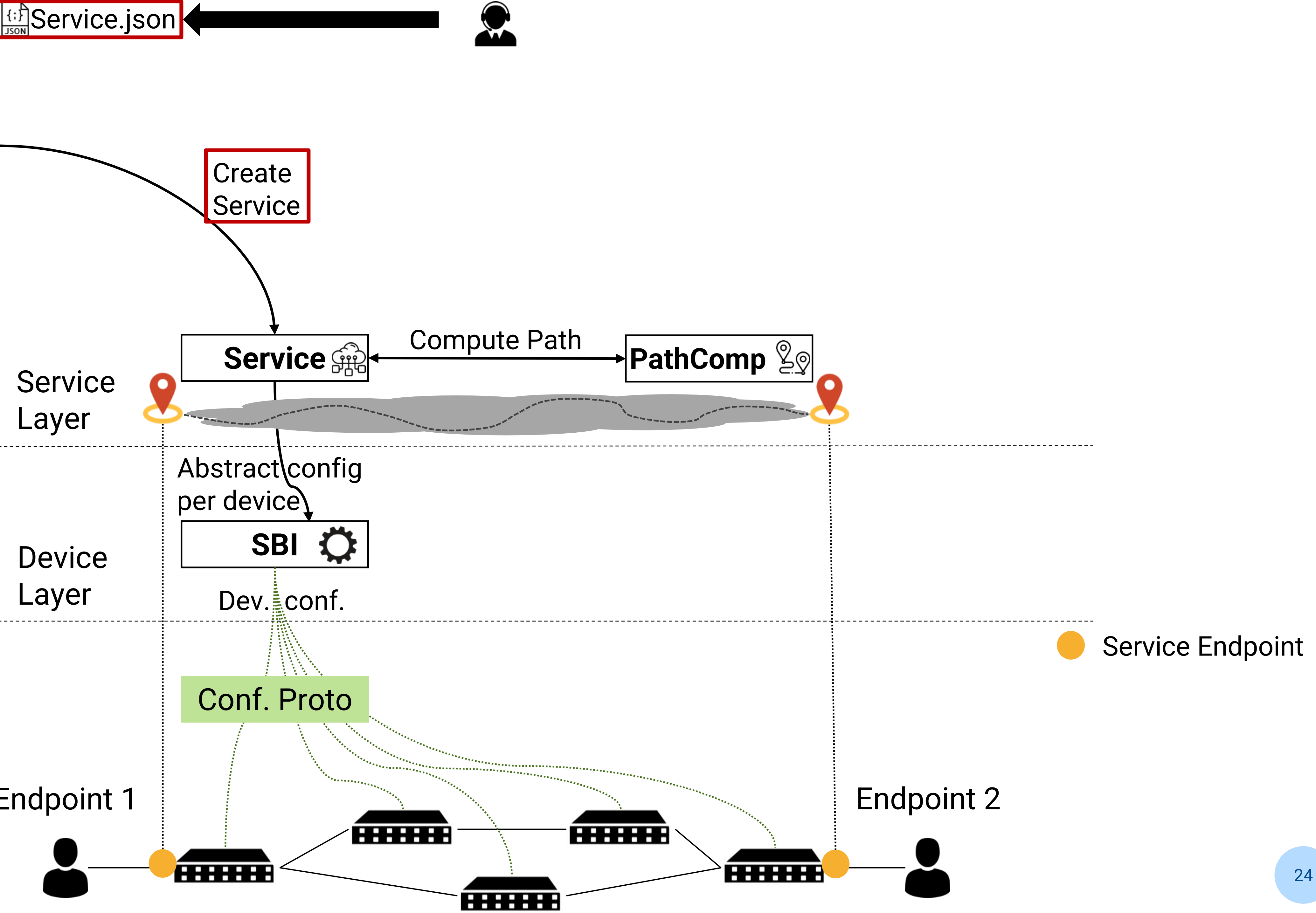
Modular path computation between endpoints

Support for heterogeneous device drivers

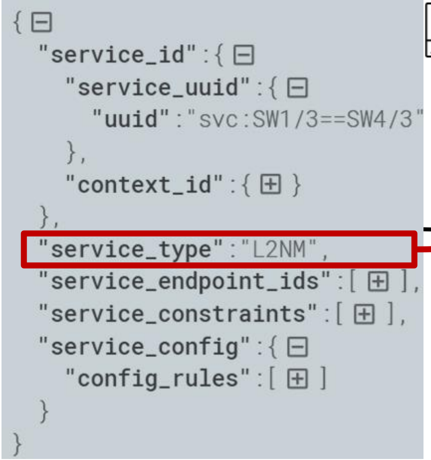




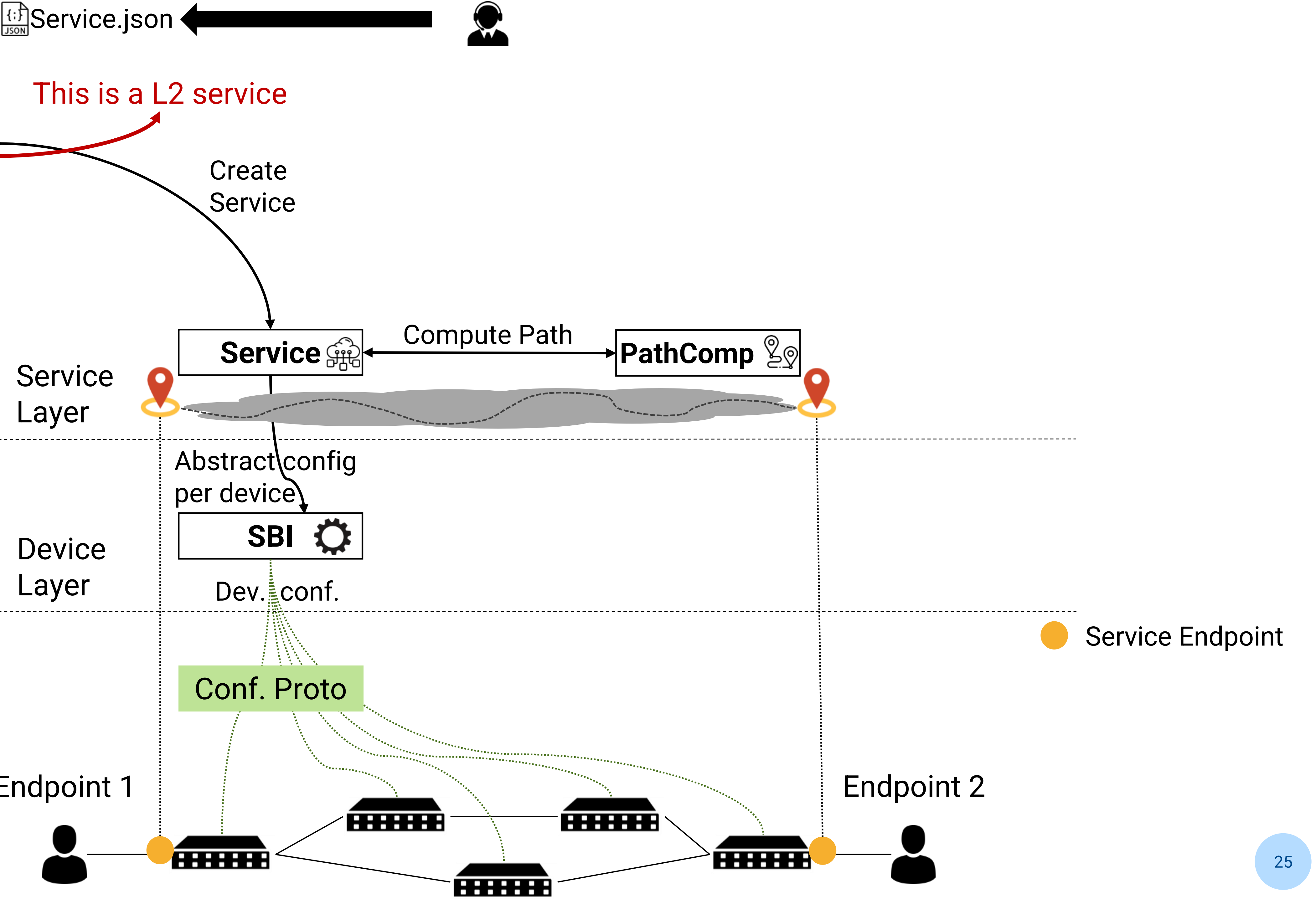
```
{ ⊟
  "service_id":{ ⊟
    "service_uuid":{ ⊟
      "uuid":"svc:SW1/3==SW4/3
    },
    "context_id":{ ⊞ }
  },
  "service_type":"L2NM",
  "service_endpoint_ids":[ ⊞ ],
  "service_constraints":[ ⊞ ],
  "service_config":{ ⊟
    "config_rules":[ ⊞ ]
  }
}
```



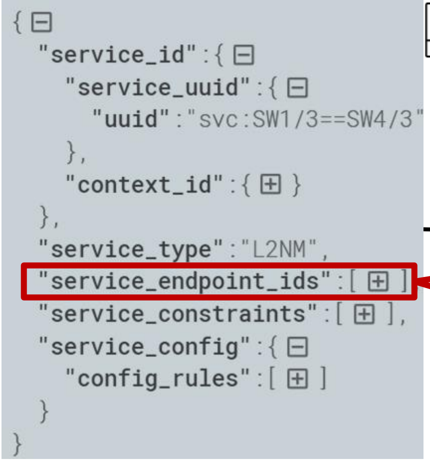
TeraFlow SDN by ETSI



```
{ ⊟
  "service_id" : { ⊟
    "service_uuid" : { ⊟
      "uuid" : "svc:SW1/3==SW4/3"
    },
    "context_id" : { ⊞ }
  },
  "service_type" : "L2NM",
  "service_endpoint_ids" : [ ⊞ ],
  "service_constraints" : [ ⊞ ],
  "service_config" : { ⊟
    "config_rules" : [ ⊞ ]
  }
}
```



TeraFlow SDN by ETSI

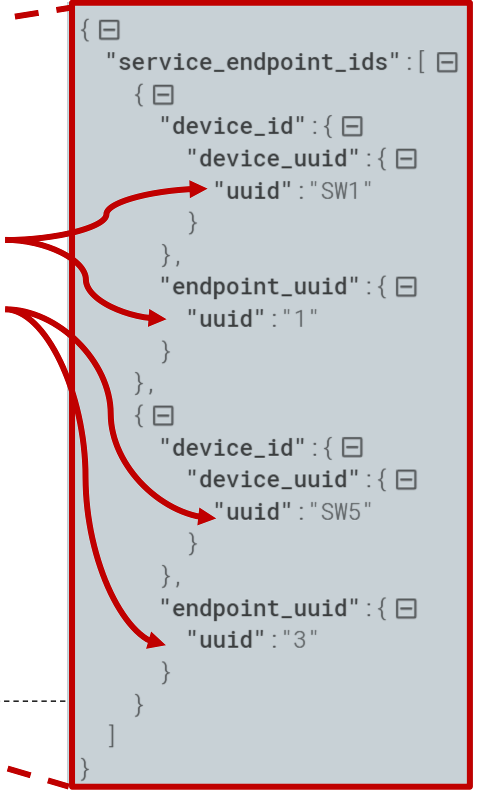


```
{ ⊟
  "service_id" : { ⊟
    "service_uuid" : { ⊟
      "uuid" : "svc:SW1/3==SW4/3"
    },
    "context_id" : { ⊞ }
  },
  "service_type" : "L2NM",
  "service_endpoint_ids" : [ ⊞ ]
  "service_constraints" : [ ⊞ ],
  "service_config" : { ⊟
    "config_rules" : [ ⊞ ]
  }
}
```

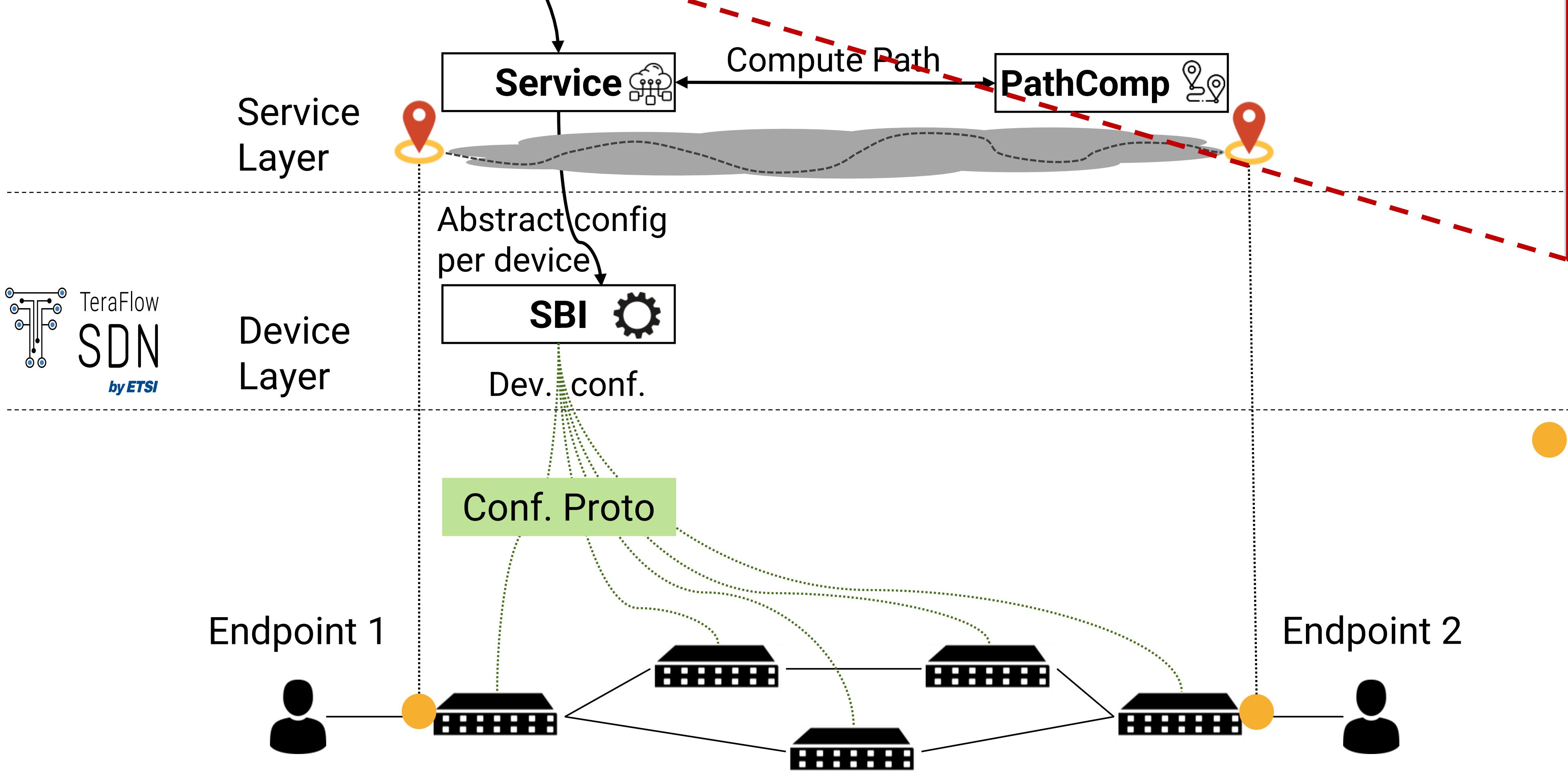
Service.json

Service begins at: SW1-P1

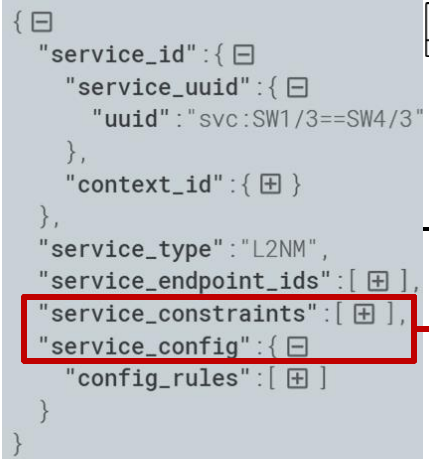
Service terminates at: SW5-P3



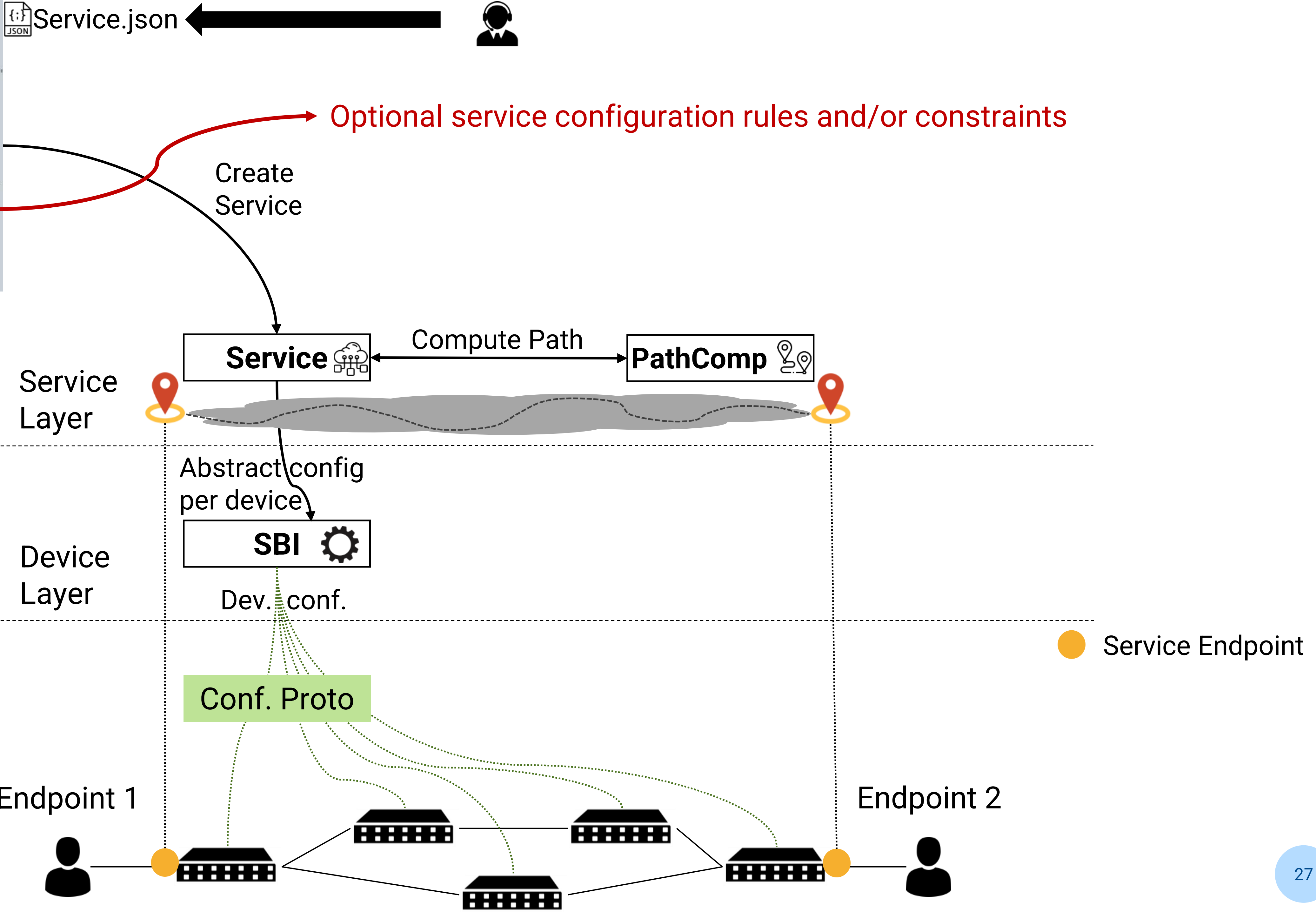
```
{ ⊟
  "service_endpoint_ids" : [ ⊟
    { ⊟
      "device_id" : { ⊟
        "device_uuid" : { ⊟
          "uuid" : "SW1"
        }
      },
      "endpoint_uuid" : { ⊟
        "uuid" : "1"
      }
    },
    { ⊟
      "device_id" : { ⊟
        "device_uuid" : { ⊟
          "uuid" : "SW5"
        }
      },
      "endpoint_uuid" : { ⊟
        "uuid" : "3"
      }
    }
  ]
}
```



● Service Endpoint



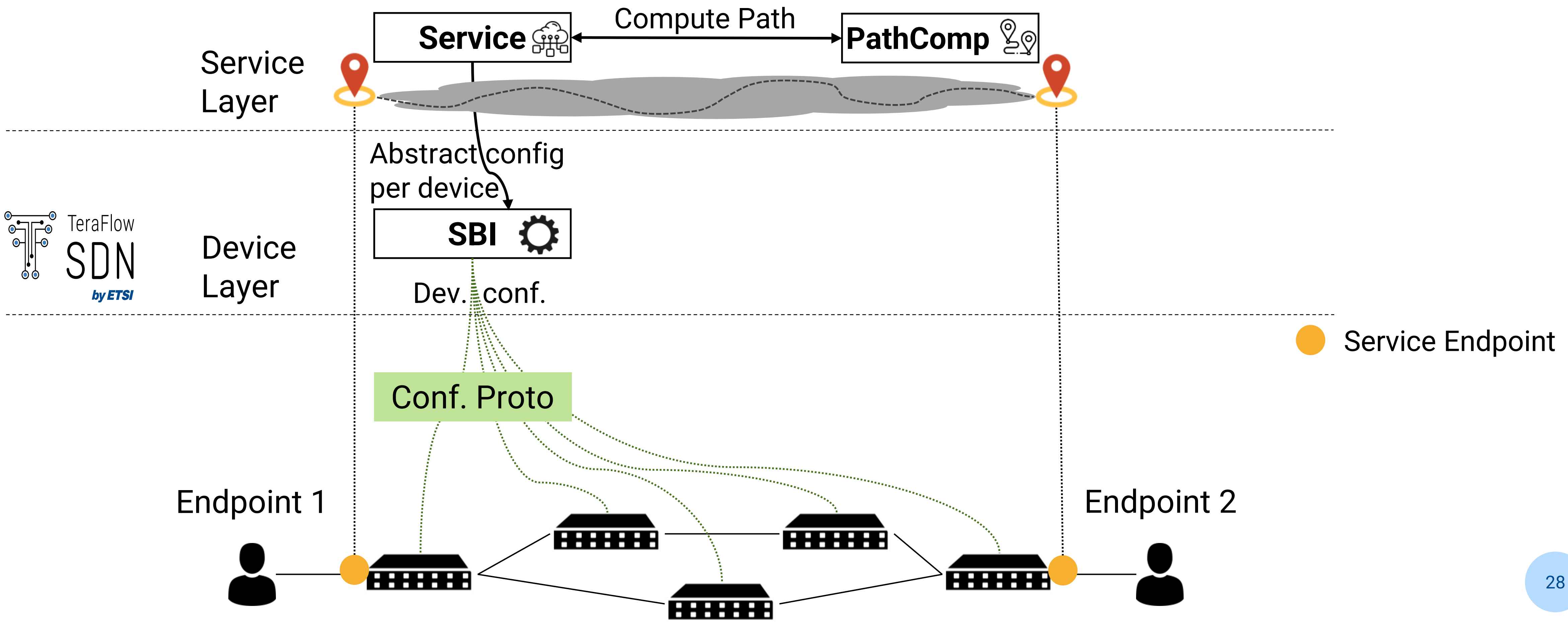
```
{ ⊟
  "service_id":{ ⊟
    "service_uuid":{ ⊟
      "uuid":"svc:SW1/3==SW4/3"
    },
    "context_id":{ ⊞ }
  },
  "service_type":"L2NM",
  "service_endpoint_ids":[ ⊞ ],
  "service_constraints":[ ⊞ ],
  "service_config":{ ⊟
    "config_rules":[ ⊞ ]
  }
}
```



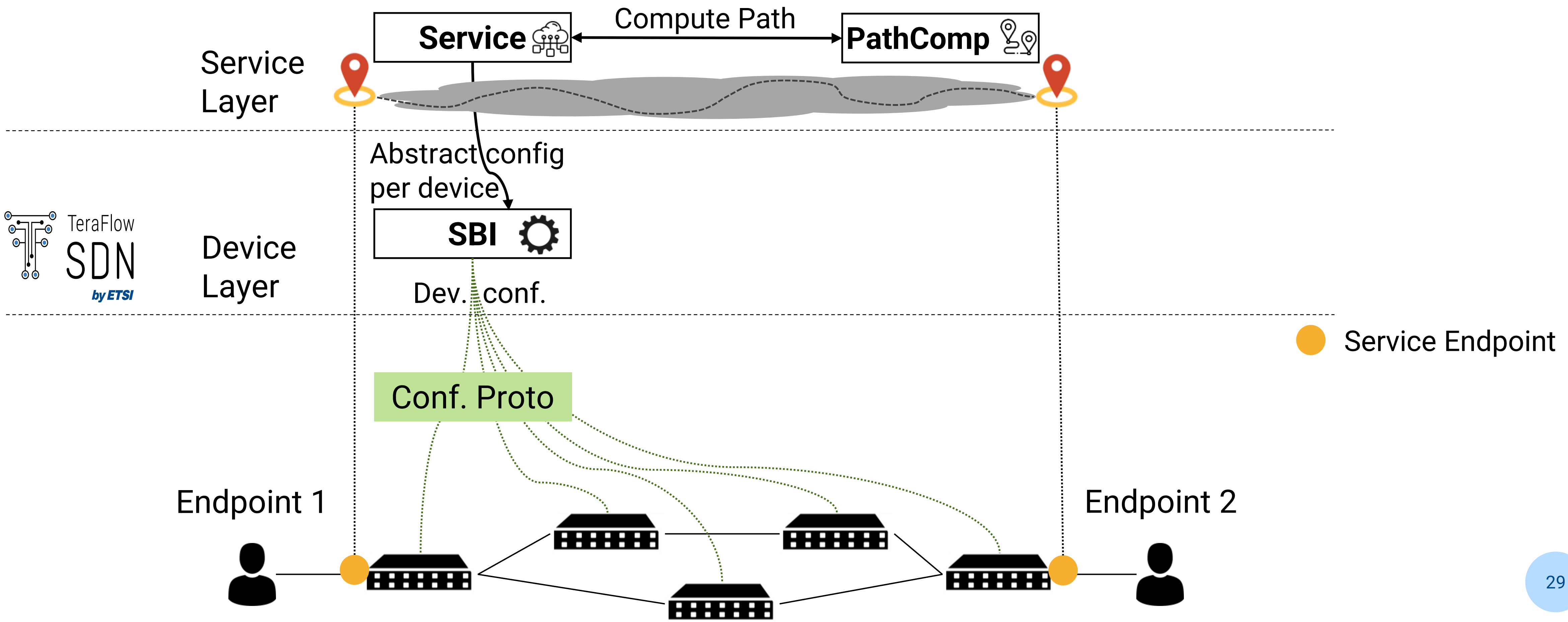
TeraFlow SDN by ETSI

The Service layer auto-translates end-to-end connectivity objectives to low level device conf./rules

Hides the complexity of the device layer

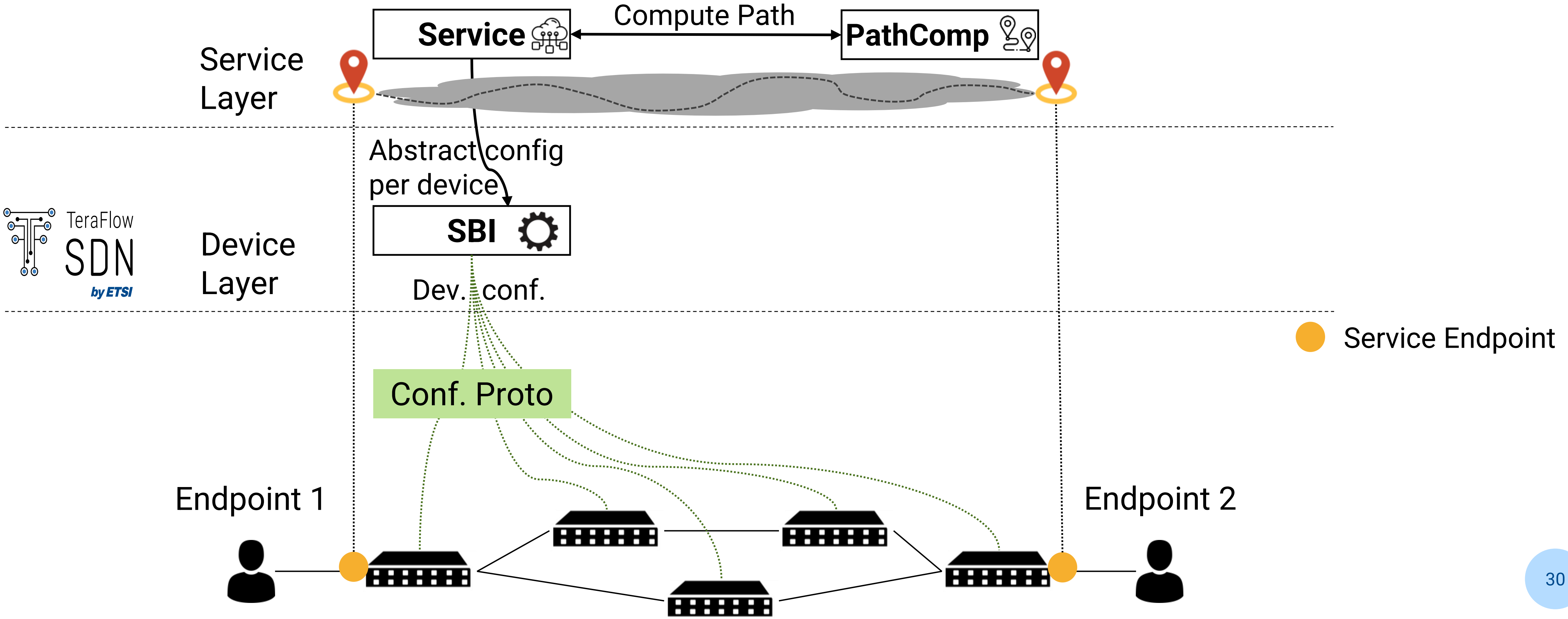


Service abstractions are key, but still not enough



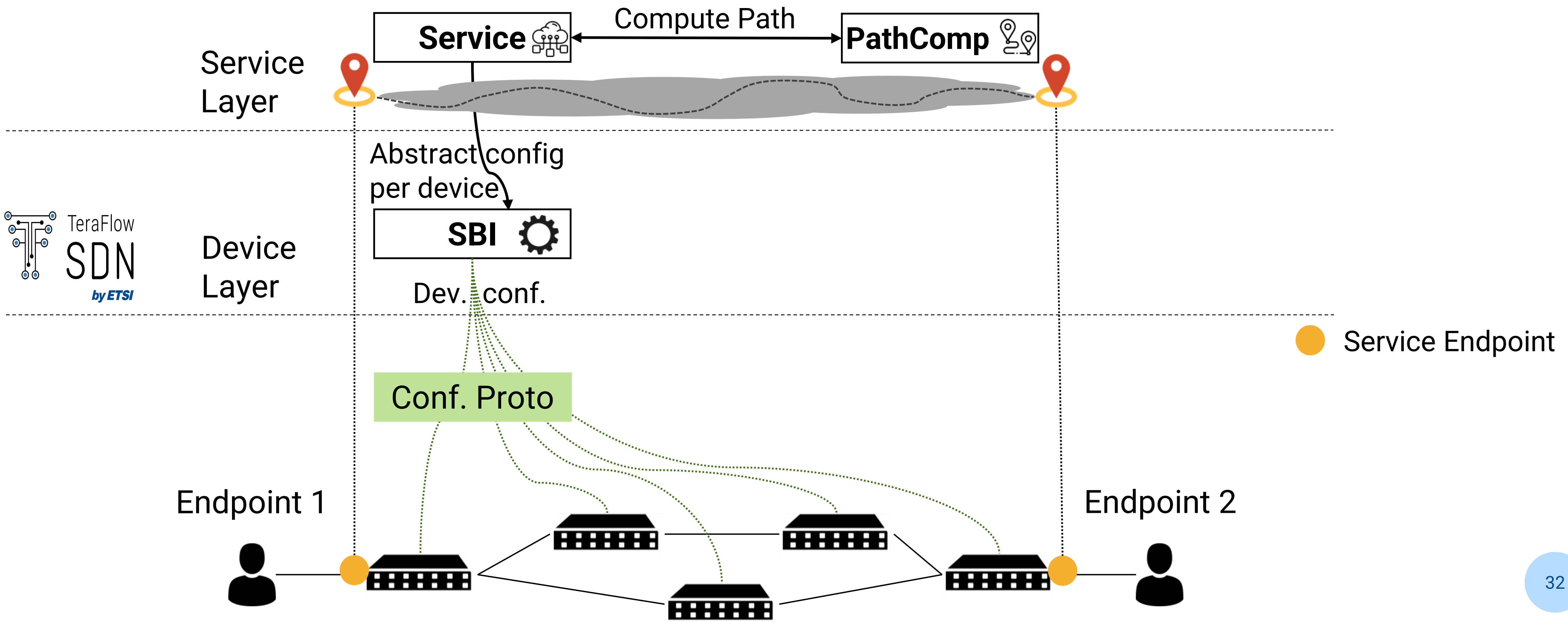
Network operators require means to trigger service APIs in an automated fashion, based on the network state

How can we achieve this?

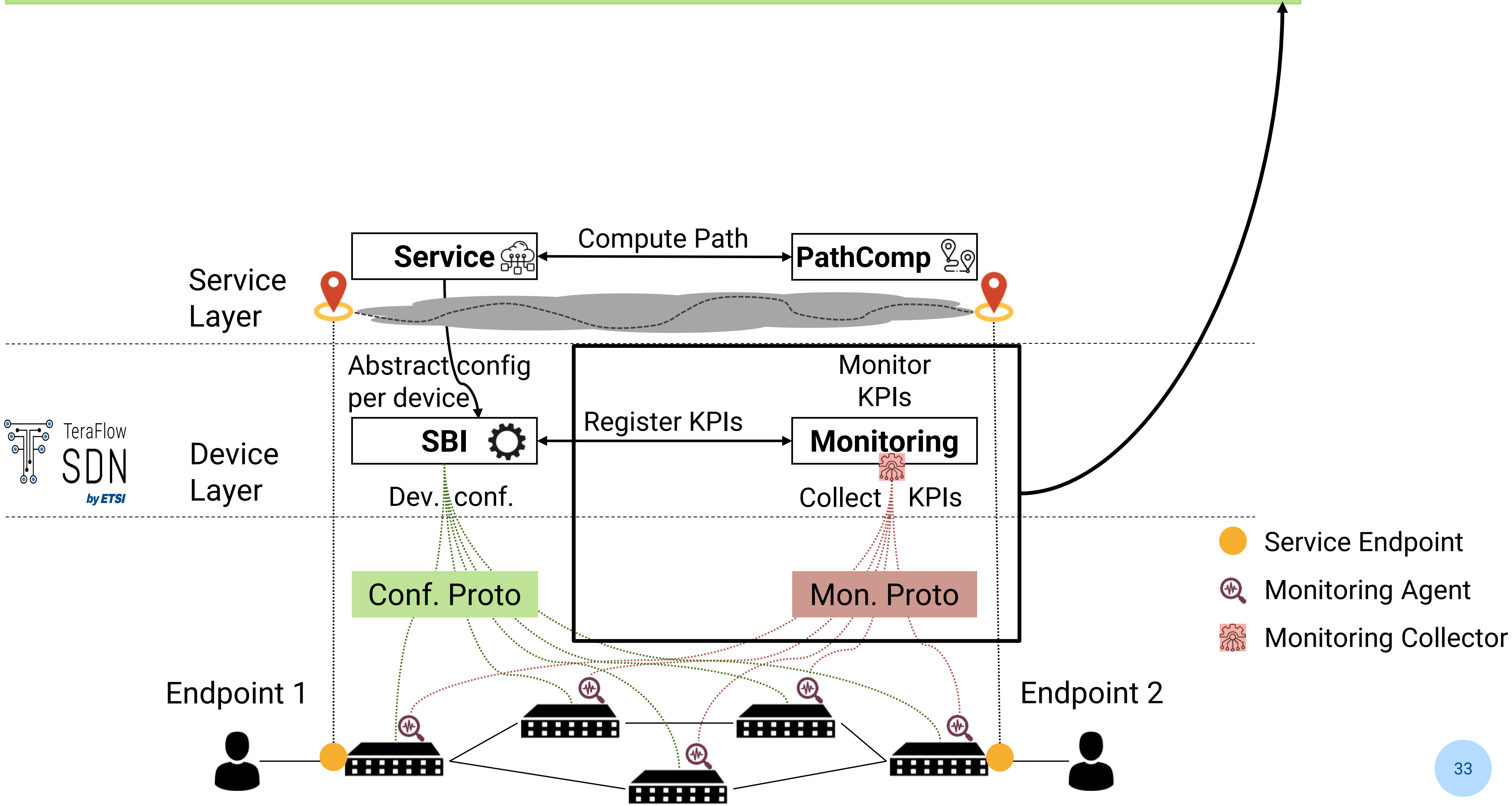


Management-level Abstractions

ETSI TFS Policy microservice

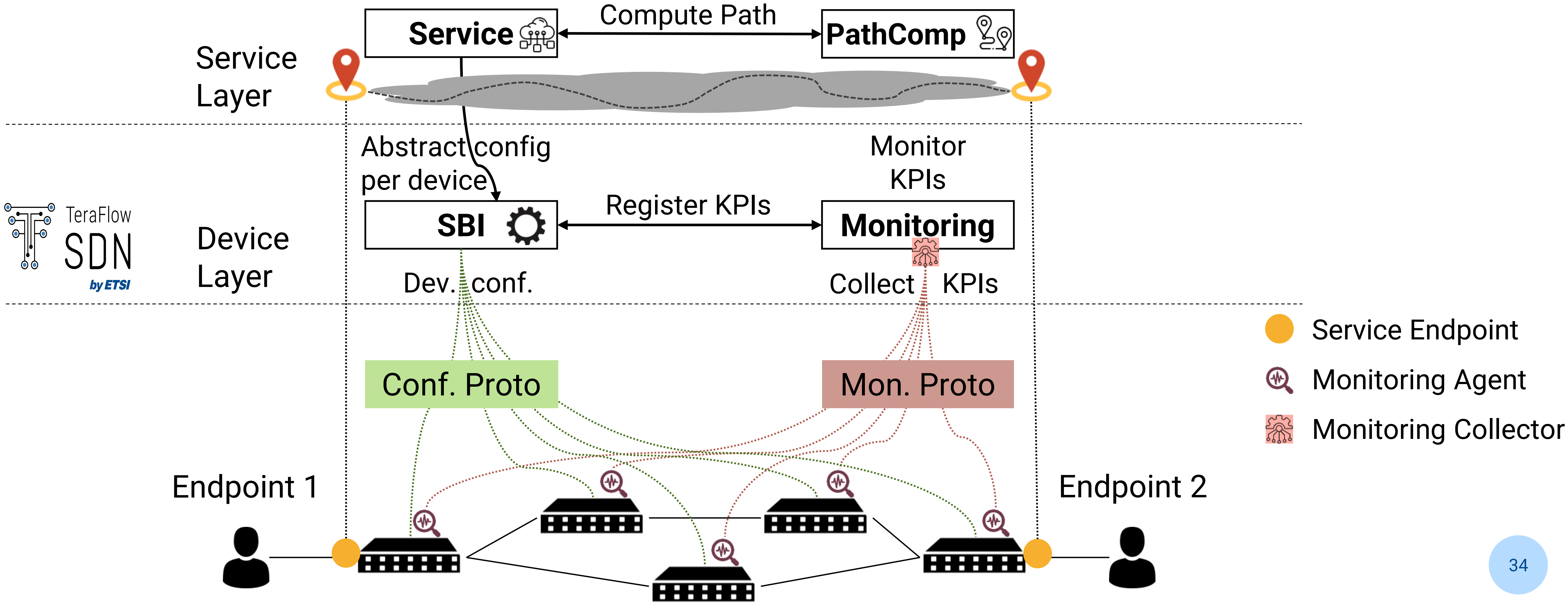

Service
Compute Path
PathComp
Service Layer
Abstract config per device
SBI
TeraFlow SDN by ETSI
Device Layer
Dev. conf.
Conf. Proto
Service Endpoint
Endpoint 1
Endpoint 2

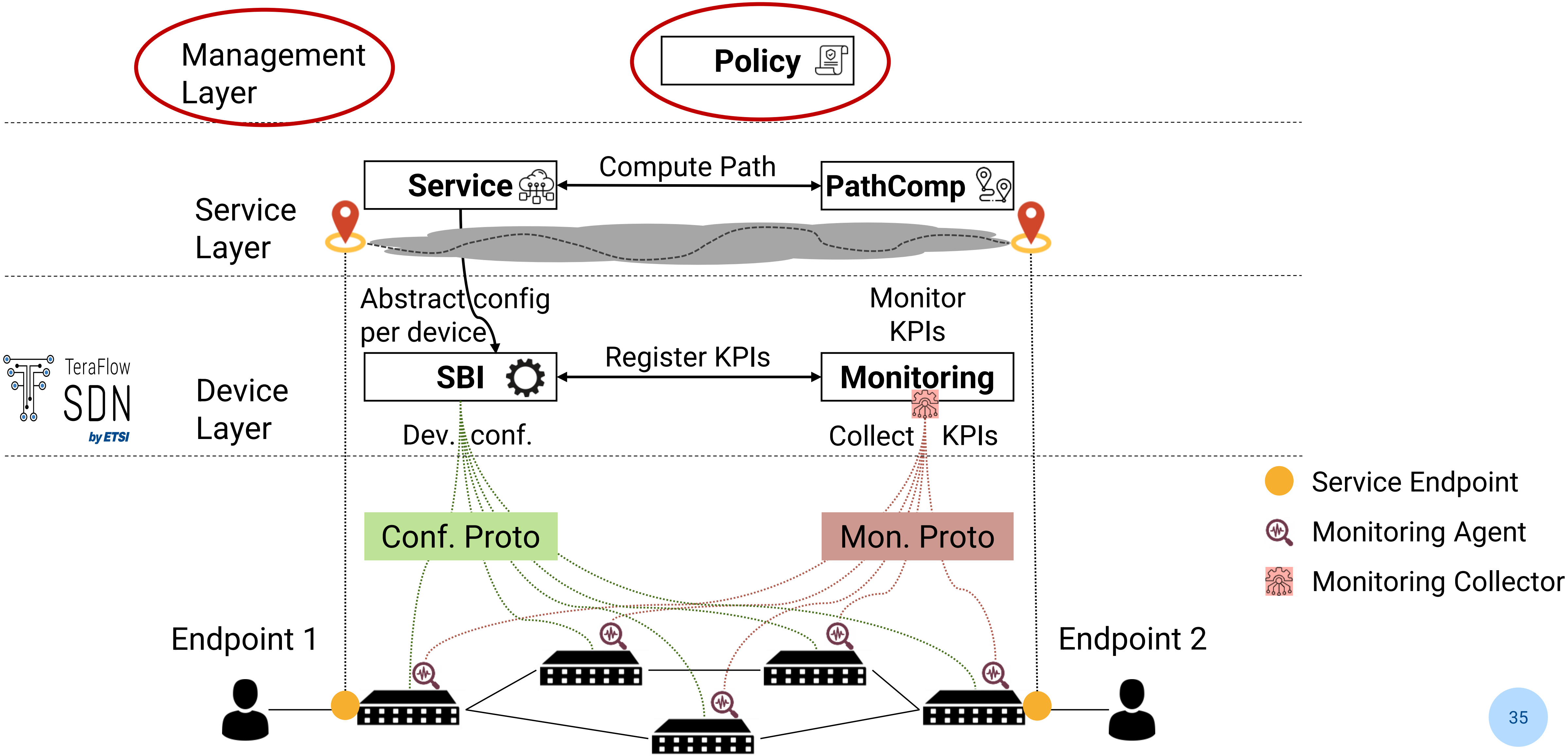
Network observability via device and service monitoring

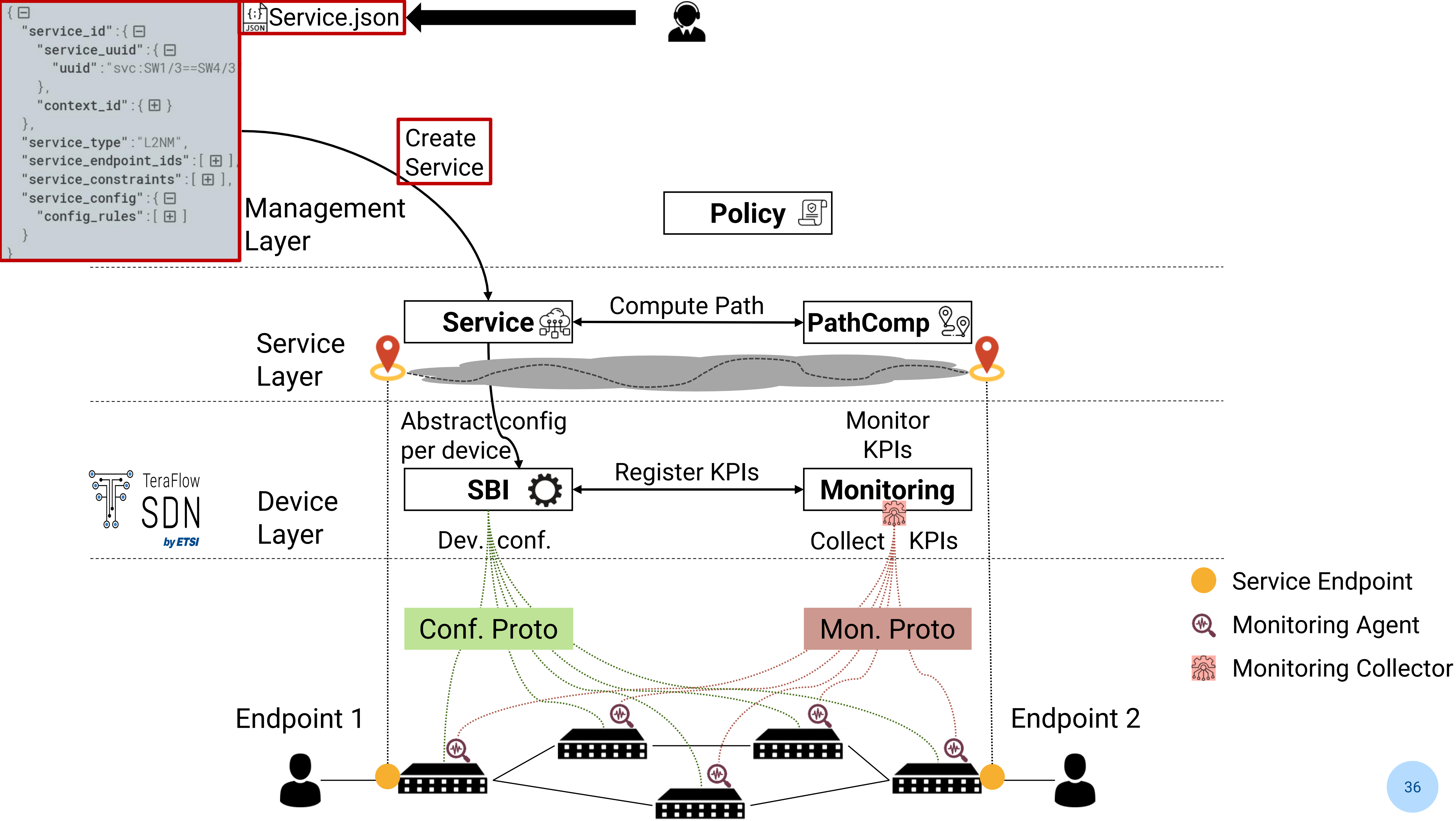


Now it is possible to capture network state!

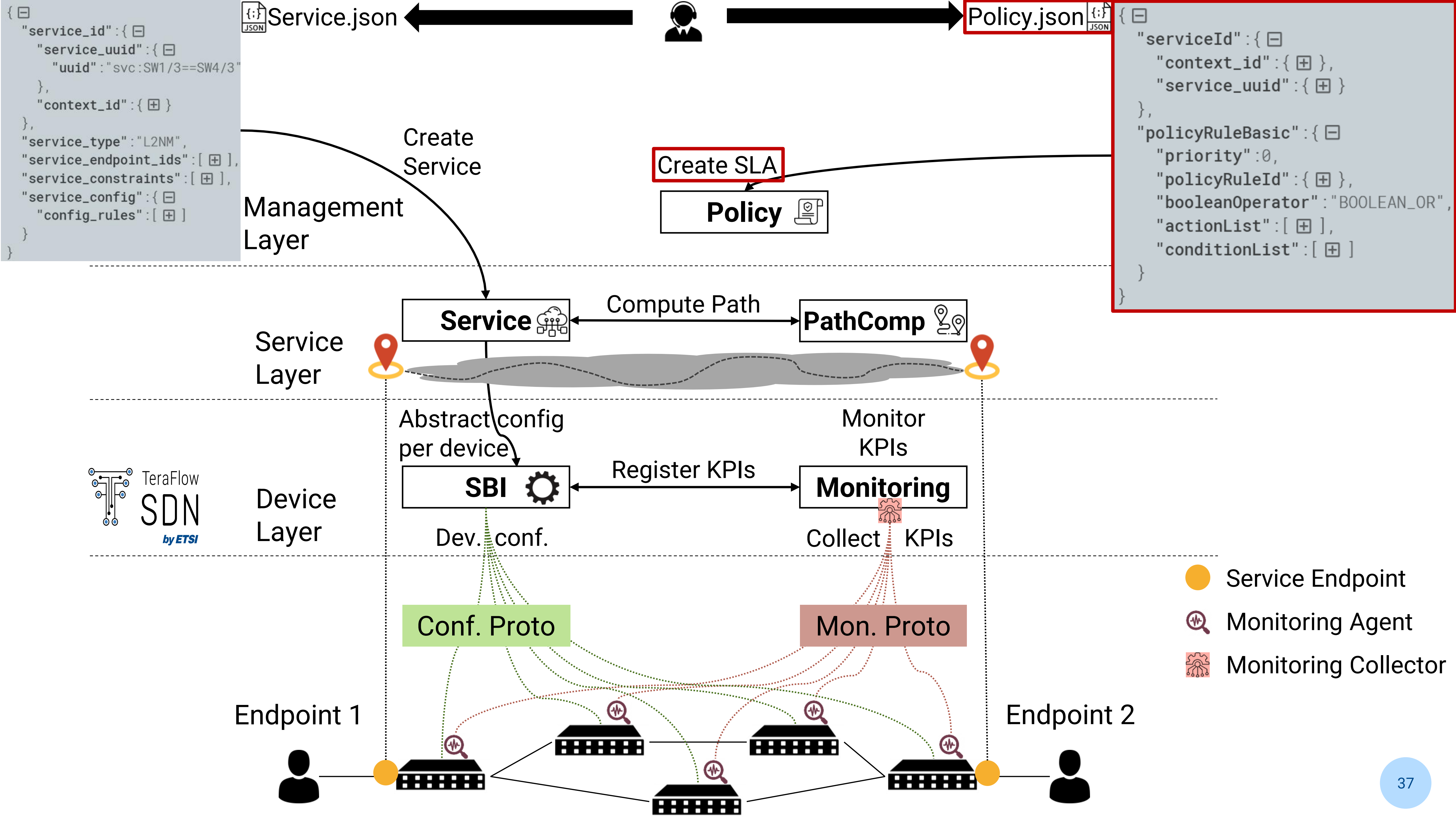
Can we exploit the synergy between Service & Monitoring?




Management Layer
Policy
Service Layer
Service
Compute Path
PathComp
Abstract config per device
Monitor KPIs
TeraFlow SDN by ETSI
Device Layer
SBI
Register KPIs
Monitoring
Dev. conf.
Collect KPIs
Conf. Proto
Mon. Proto
Endpoint 1
Endpoint 2
Service Endpoint
Monitoring Agent
Monitoring Collector

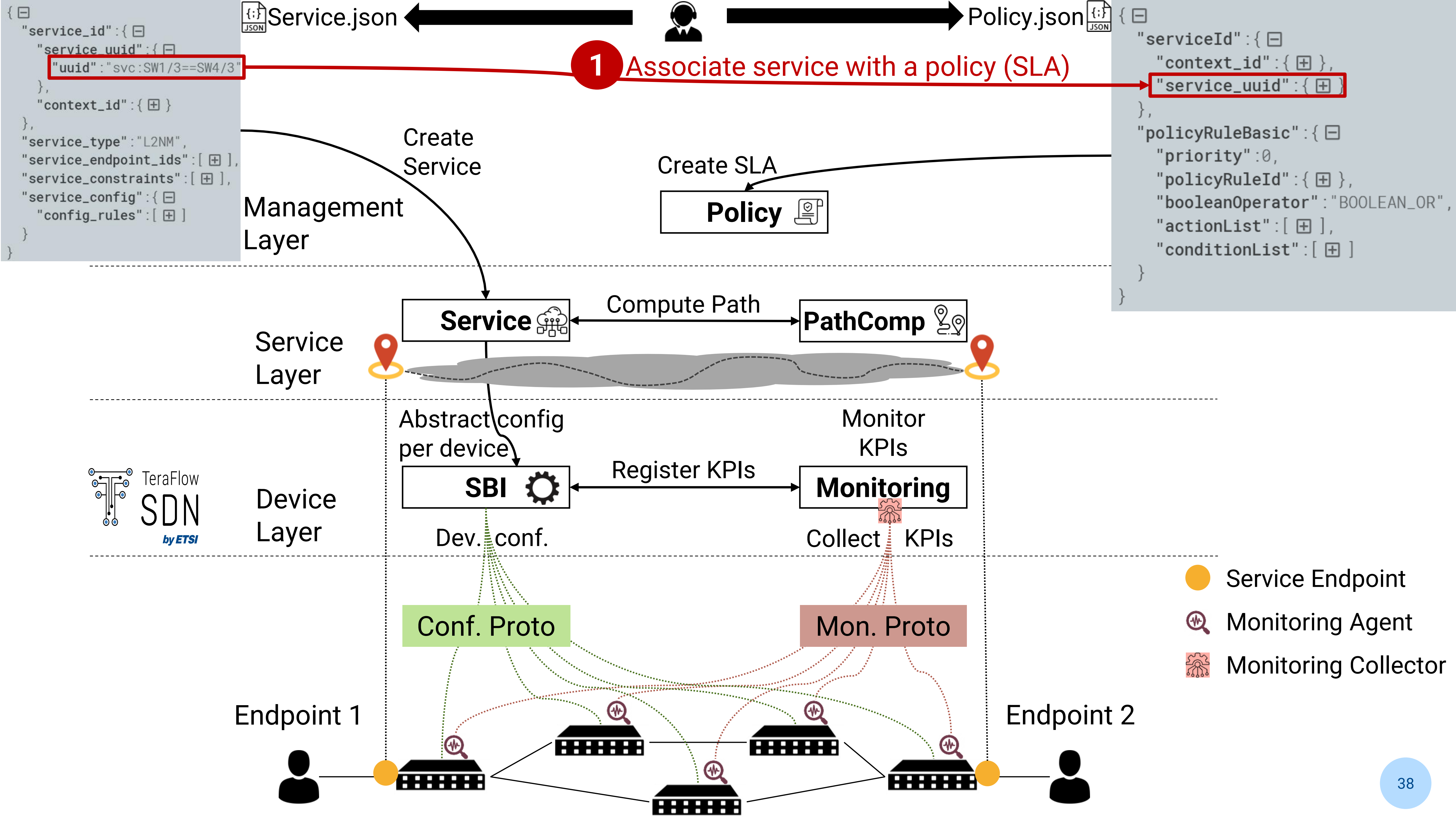


```
{ ⊟
  "service_id":{ ⊟
    "service_uuid":{ ⊟
      "uuid":"svc:SW1/3==SW4/3
    },
    "context_id":{ ⊞ }
  },
  "service_type":"L2NM",
  "service_endpoint_ids":[ ⊞ ],
  "service_constraints":[ ⊞ ],
  "service_config":{ ⊟
    "config_rules":[ ⊞ ]
  }
}
```



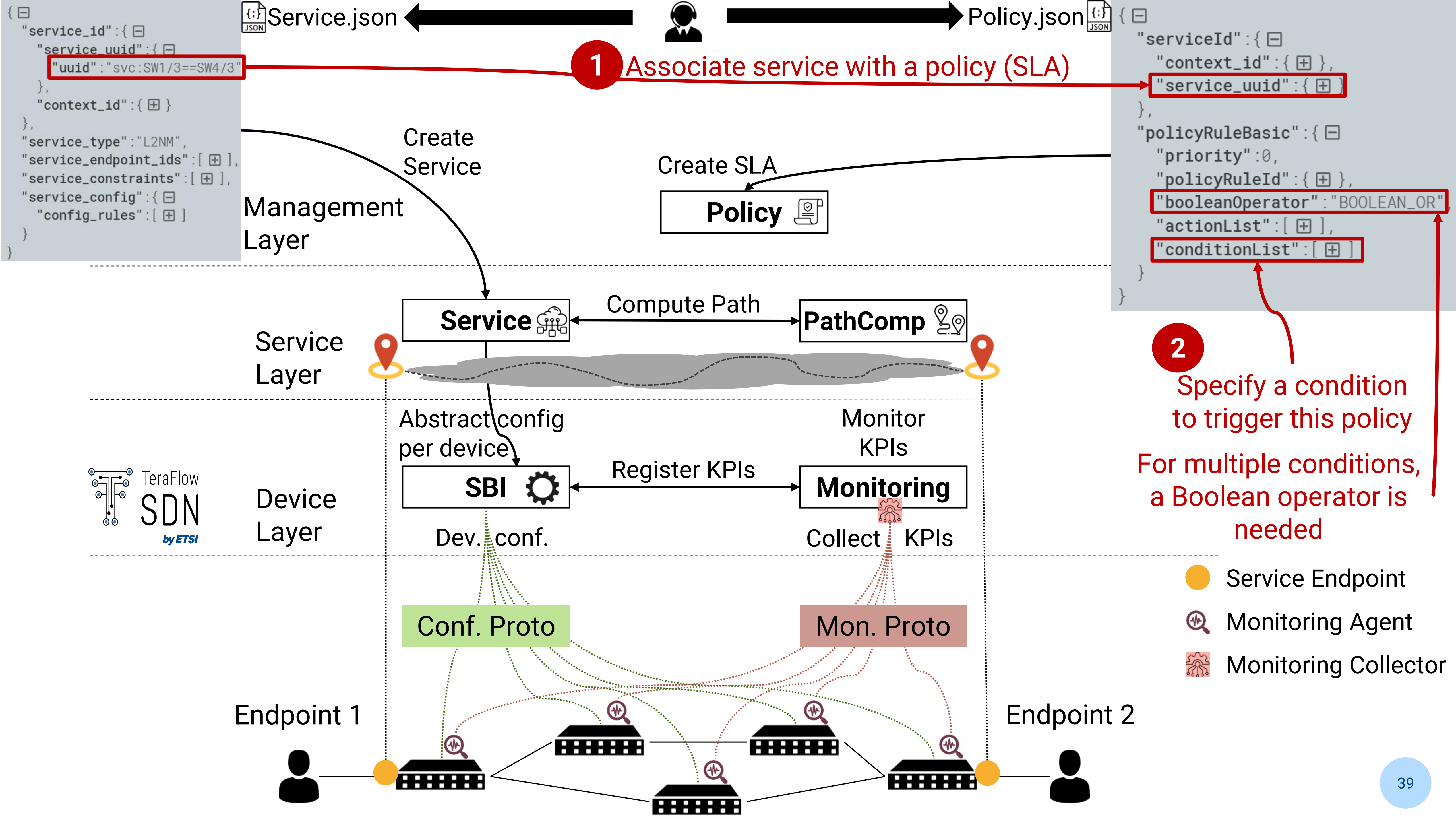
```
{ ⊟
  "service_id":{ ⊟
    "service_uuid":{ ⊟
      "uuid":"svc:SW1/3==SW4/3"
    },
    "context_id":{ ⊞ }
  },
  "service_type":"L2NM",
  "service_endpoint_ids":[ ⊞ ],
  "service_constraints":[ ⊞ ],
  "service_config":{ ⊟
    "config_rules":[ ⊞ ]
  }
}
```

```
{ ⊟
  "serviceId":{ ⊟
    "context_id":{ ⊞ },
    "service_uuid":{ ⊞ }
  },
  "policyRuleBasic":{ ⊟
    "priority":0,
    "policyRuleId":{ ⊞ },
    "booleanOperator":"BOOLEAN_OR",
    "actionList":[ ⊞ ],
    "conditionList":[ ⊞ ]
  }
}
```



```
{ ⊟
  "service_id" : { ⊟
    "service_uuid" : { ⊟
      "uuid" : "svc:SW1/3==SW4/3"
    },
    "context_id" : { ⊞ }
  },
  "service_type" : "L2NM",
  "service_endpoint_ids" : [ ⊞ ],
  "service_constraints" : [ ⊞ ],
  "service_config" : { ⊟
    "config_rules" : [ ⊞ ]
  }
}
```

```
{ ⊟
  "serviceId" : { ⊟
    "context_id" : { ⊞ },
    "service_uuid" : { ⊞ }
  },
  "policyRuleBasic" : { ⊟
    "priority" : 0,
    "policyRuleId" : { ⊞ },
    "booleanOperator" : "BOOLEAN_OR",
    "actionList" : [ ⊞ ],
    "conditionList" : [ ⊞ ]
  }
}
```



Service.json

```
{ ⊟
  "service_id" : { ⊟
    "service_uuid" : { ⊟
      "uuid" : "svc:SW1/3==SW4/3"
    },
    "context_id" : { ⊞ }
  },
  "service_type" : "L2NM",
  "service_endpoint_ids" : [ ⊞ ],
  "service_constraints" : [ ⊞ ],
  "service_config" : { ⊟
    "config_rules" : [ ⊞ ]
  }
}
```

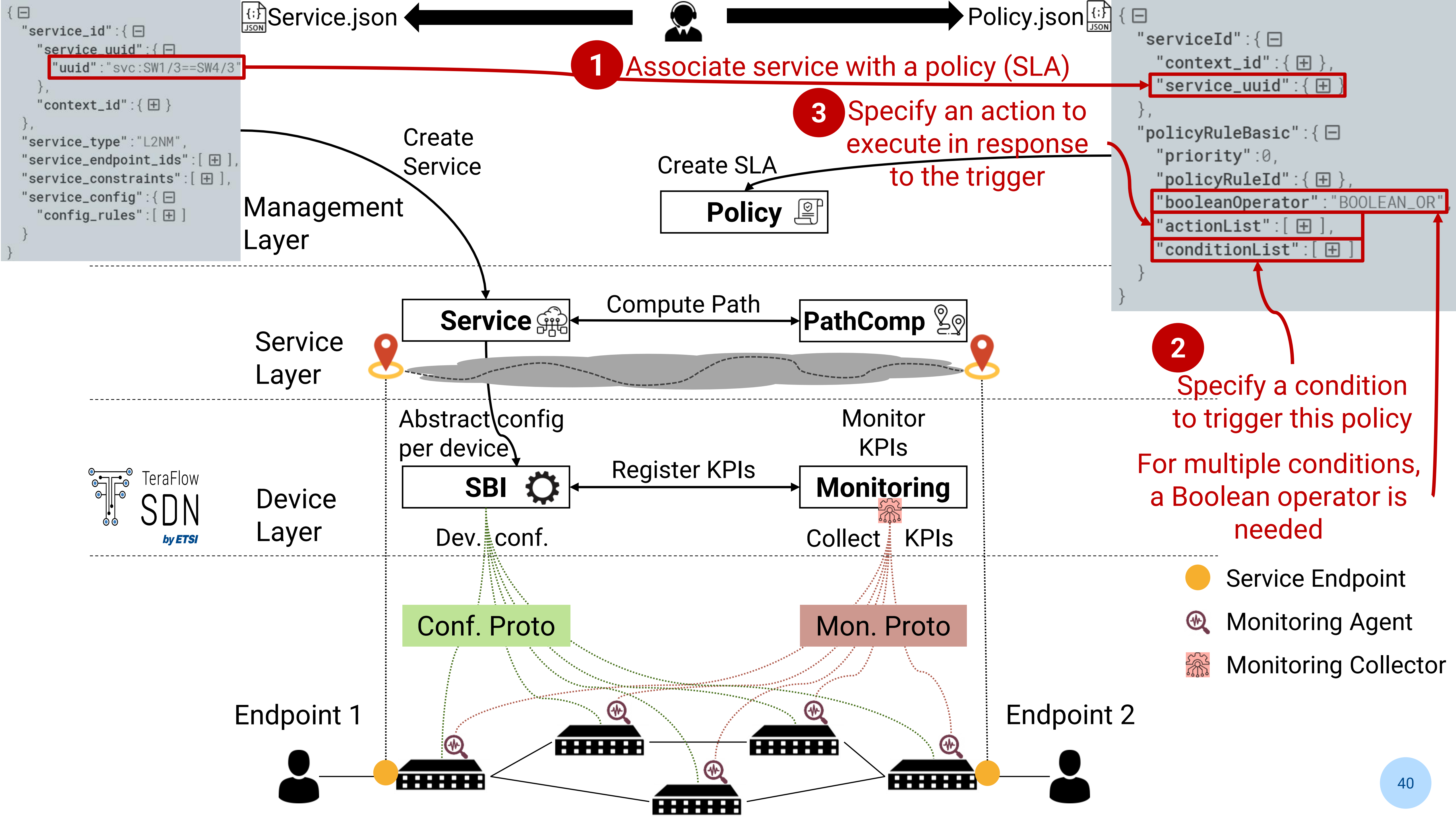
Policy.json

```
{ ⊟
  "serviceId" : { ⊟
    "context_id" : { ⊞ },
    "service_uuid" : { ⊞ }
  },
  "policyRuleBasic" : { ⊟
    "priority" : 0,
    "policyRuleId" : { ⊞ },
    "booleanOperator" : "BOOLEAN_OR",
    "actionList" : [ ⊞ ],
    "conditionList" : [ ⊞ ]
  }
}
```

1 Associate service with a policy (SLA)

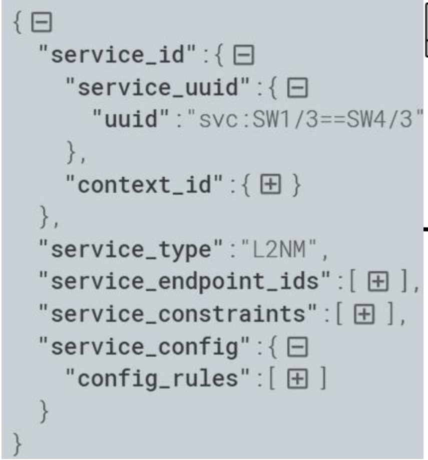
2 Specify a condition to trigger this policy

For multiple conditions, a Boolean operator is needed

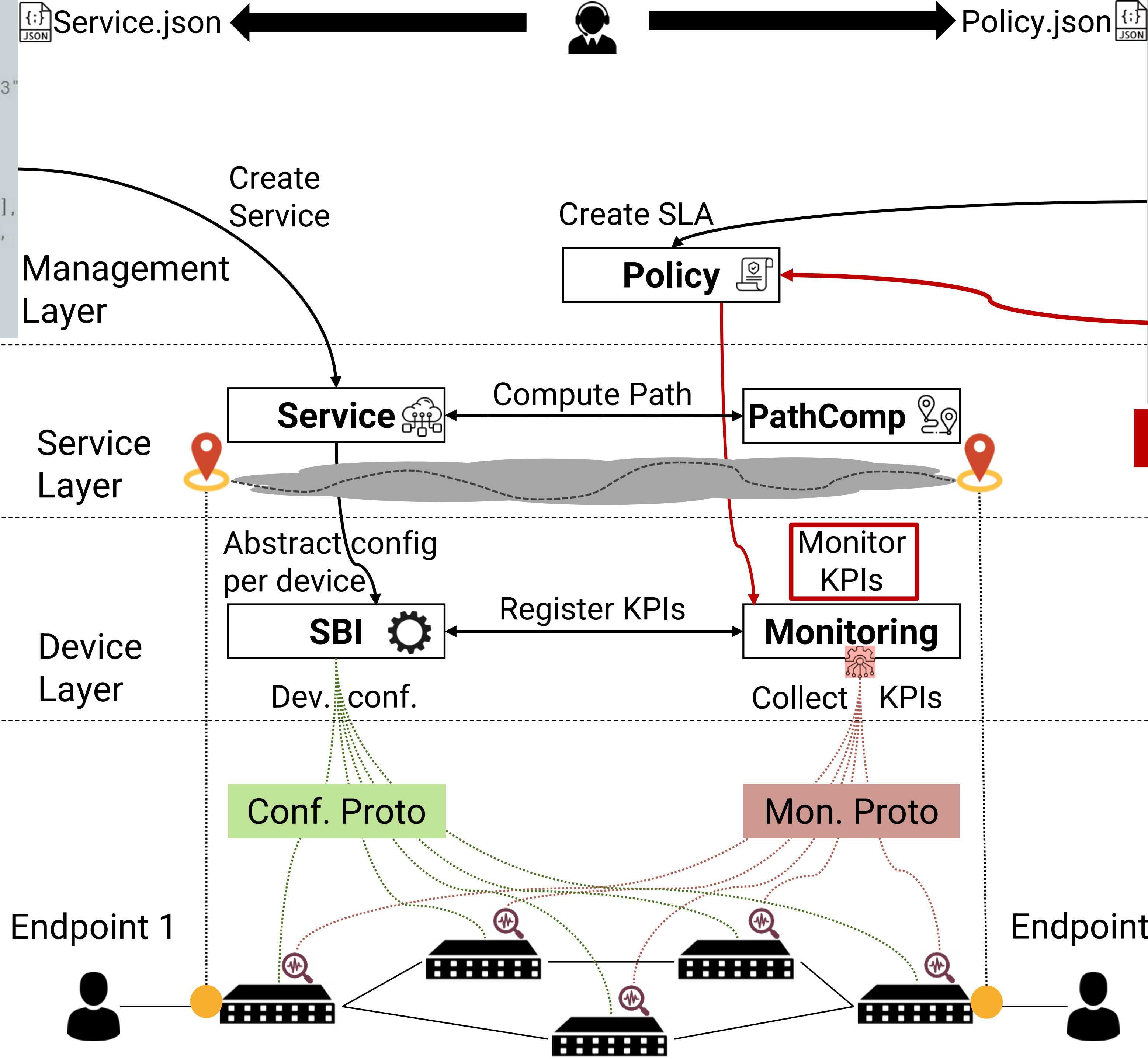


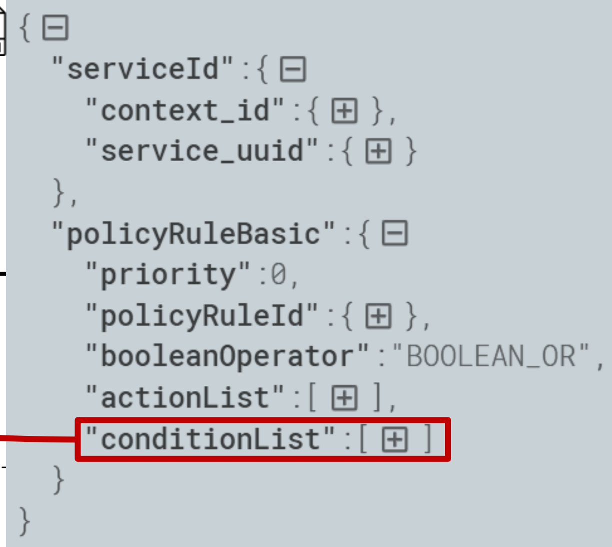
```
{ ⊟
  "service_id" : { ⊟
    "service_uuid" : { ⊟
      "uuid" : "svc:SW1/3==SW4/3"
    },
    "context_id" : { ⊞ }
  },
  "service_type" : "L2NM",
  "service_endpoint_ids" : [ ⊞ ],
  "service_constraints" : [ ⊞ ],
  "service_config" : { ⊟
    "config_rules" : [ ⊞ ]
  }
}
```

```
{ ⊟
  "serviceId" : { ⊟
    "context_id" : { ⊞ },
    "service_uuid" : { ⊞ }
  },
  "policyRuleBasic" : { ⊟
    "priority" : 0,
    "policyRuleId" : { ⊞ },
    "booleanOperator" : "BOOLEAN_OR",
    "actionList" : [ ⊞ ],
    "conditionList" : [ ⊞ ]
  }
}
```



```
{ ⊟
  "service_id":{ ⊟
    "service_uuid":{ ⊟
      "uuid":"svc:SW1/3==SW4/3"
    },
    "context_id":{ ⊞ }
  },
  "service_type":"L2NM",
  "service_endpoint_ids":[ ⊞ ],
  "service_constraints":[ ⊞ ],
  "service_config":{ ⊟
    "config_rules":[ ⊞ ]
  }
}
```





```
{ ⊟
  "serviceId":{ ⊟
    "context_id":{ ⊞ },
    "service_uuid":{ ⊞ }
  },
  "policyRuleBasic":{ ⊟
    "priority":0,
    "policyRuleId":{ ⊞ },
    "booleanOperator":"BOOLEAN_OR",
    "actionList":[ ⊞ ],
    "conditionList":[ ⊞ ]
  }
}
```

Condition

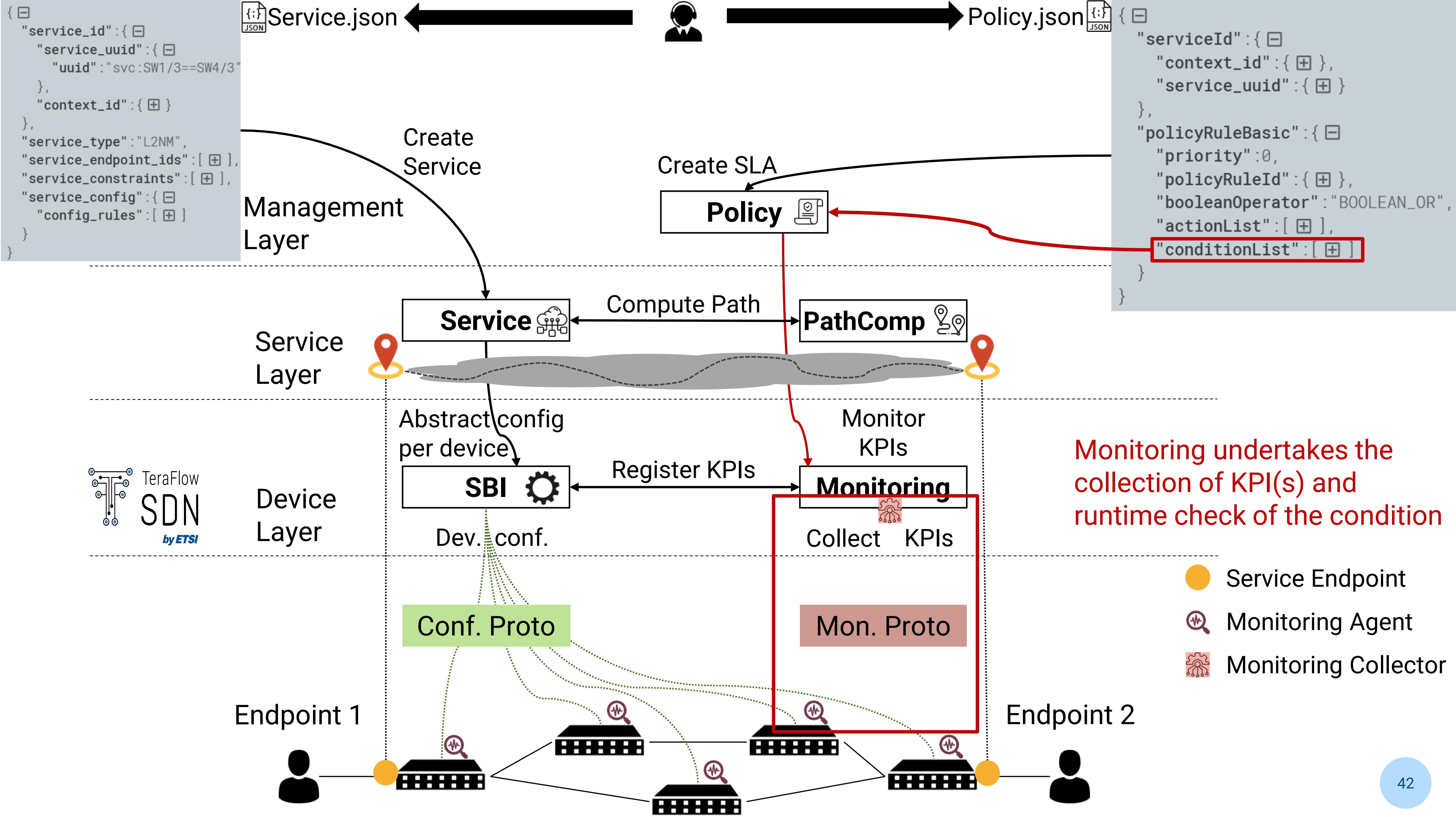
Register a new Monitoring alarm out of the KPI(s) in the condition list

TeraFlow SDN by ETSI

● Service Endpoint

Monitoring Agent

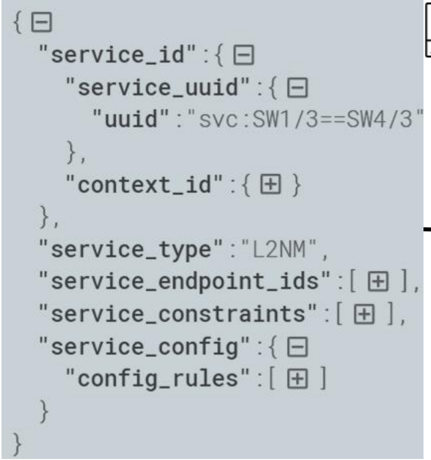
Monitoring Collector



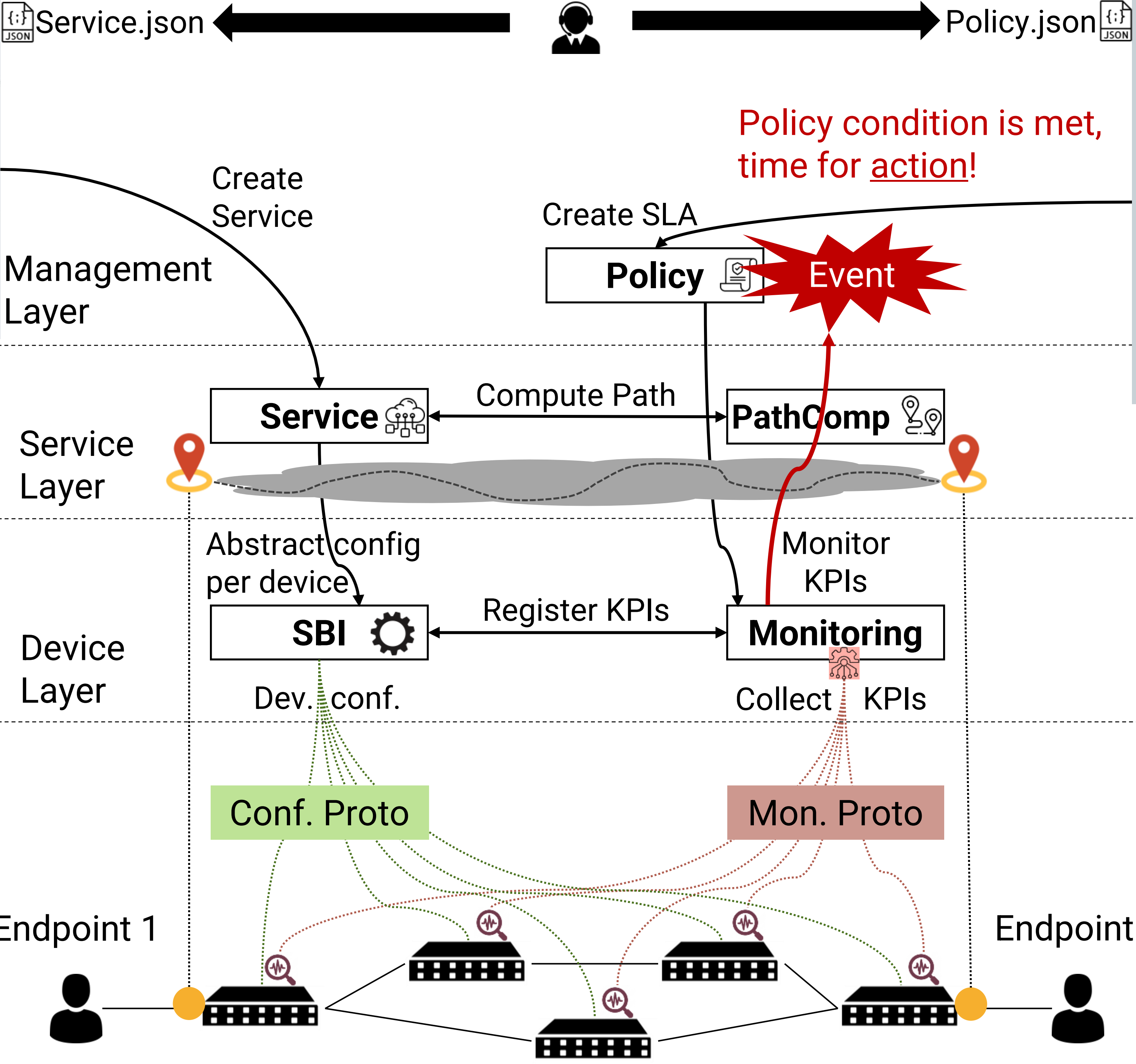
```
{ ⊟
  "service_id":{ ⊟
    "service_uuid":{ ⊟
      "uuid":"svc:SW1/3==SW4/3"
    },
    "context_id":{ ⊞ }
  },
  "service_type":"L2NM",
  "service_endpoint_ids":[ ⊞ ],
  "service_constraints":[ ⊞ ],
  "service_config":{ ⊟
    "config_rules":[ ⊞ ]
  }
}
```

```
{ ⊟
  "serviceId":{ ⊟
    "context_id":{ ⊞ },
    "service_uuid":{ ⊞ }
  },
  "policyRuleBasic":{ ⊟
    "priority":0,
    "policyRuleId":{ ⊞ },
    "booleanOperator":"BOOLEAN_OR",
    "actionList":[ ⊞ ],
    "conditionList":[ ⊞ ]
  }
}
```

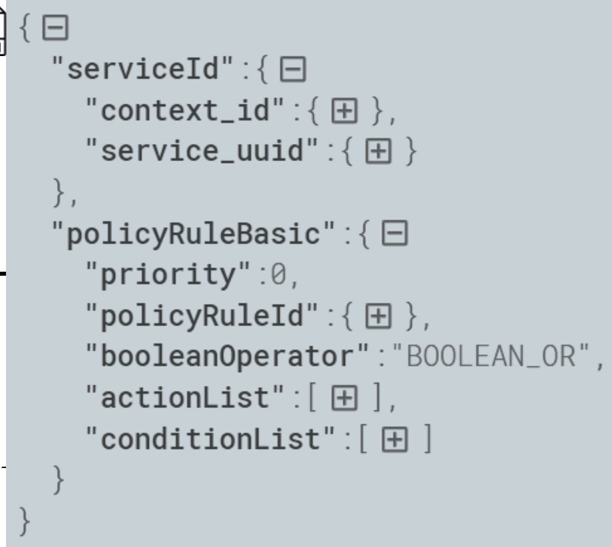
Monitoring undertakes the collection of KPI(s) and runtime check of the condition



```
{ ⊟
  "service_id":{ ⊟
    "service_uuid":{ ⊟
      "uuid":"svc:SW1/3==SW4/3"
    },
    "context_id":{ ⊞ }
  },
  "service_type":"L2NM",
  "service_endpoint_ids":[ ⊞ ],
  "service_constraints":[ ⊞ ],
  "service_config":{ ⊟
    "config_rules":[ ⊞ ]
  }
}
```



Policy condition is met, time for action!



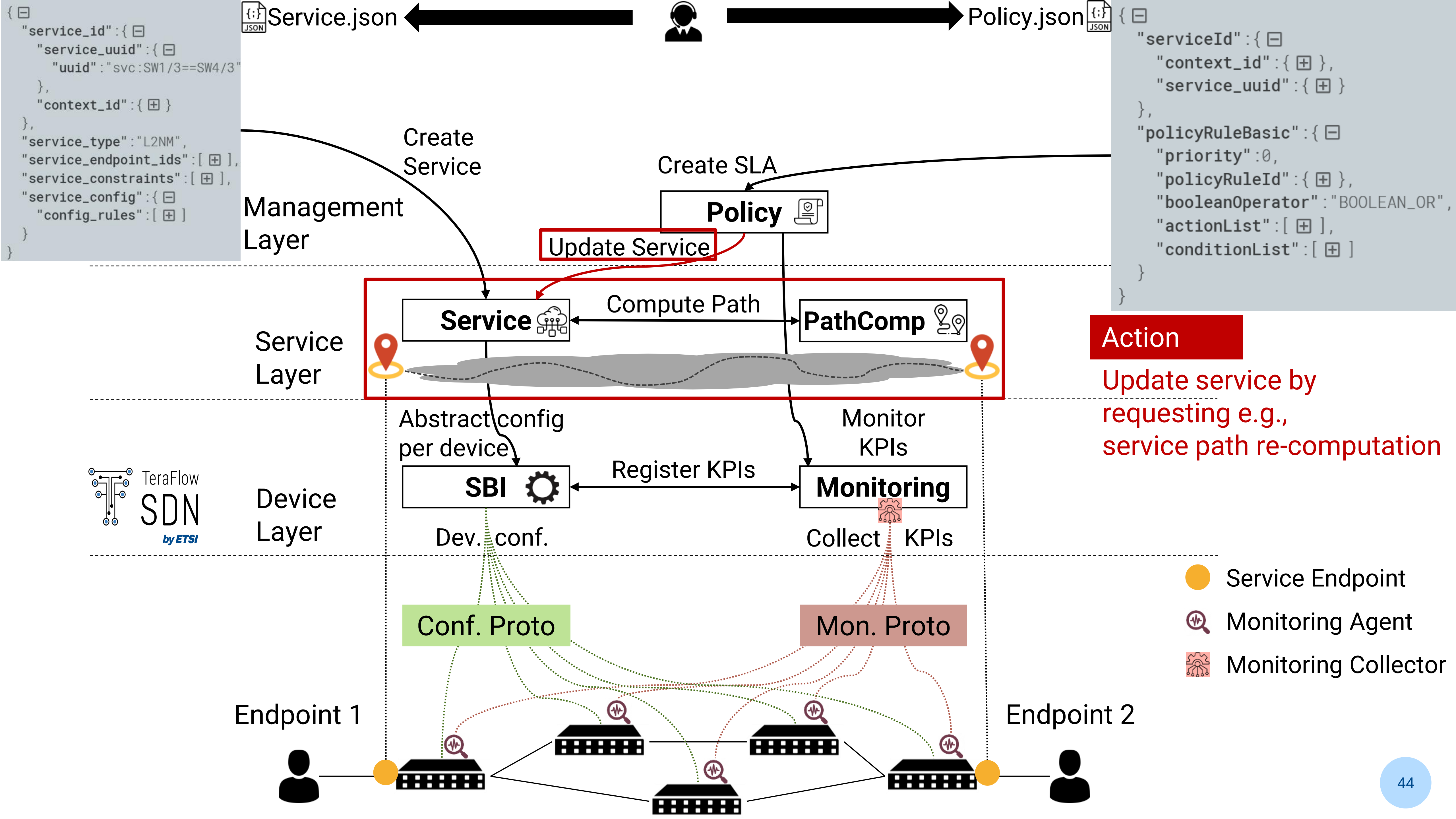
```
{ ⊟
  "serviceId":{ ⊟
    "context_id":{ ⊞ },
    "service_uuid":{ ⊞ }
  },
  "policyRuleBasic":{ ⊟
    "priority":0,
    "policyRuleId":{ ⊞ },
    "booleanOperator":"BOOLEAN_OR",
    "actionList":[ ⊞ ],
    "conditionList":[ ⊞ ]
  }
}
```

TeraFlow SDN by ETSI

● Service Endpoint

Monitoring Agent

Monitoring Collector

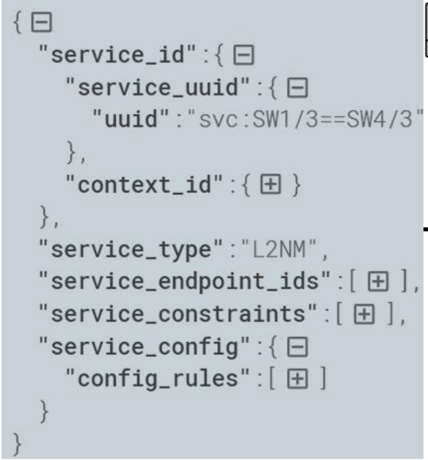


```
{ ⊟
  "service_id":{ ⊟
    "service_uuid":{ ⊟
      "uuid":"svc:SW1/3==SW4/3"
    },
    "context_id":{ ⊞ }
  },
  "service_type":"L2NM",
  "service_endpoint_ids":[ ⊞ ],
  "service_constraints":[ ⊞ ],
  "service_config":{ ⊟
    "config_rules":[ ⊞ ]
  }
}
```

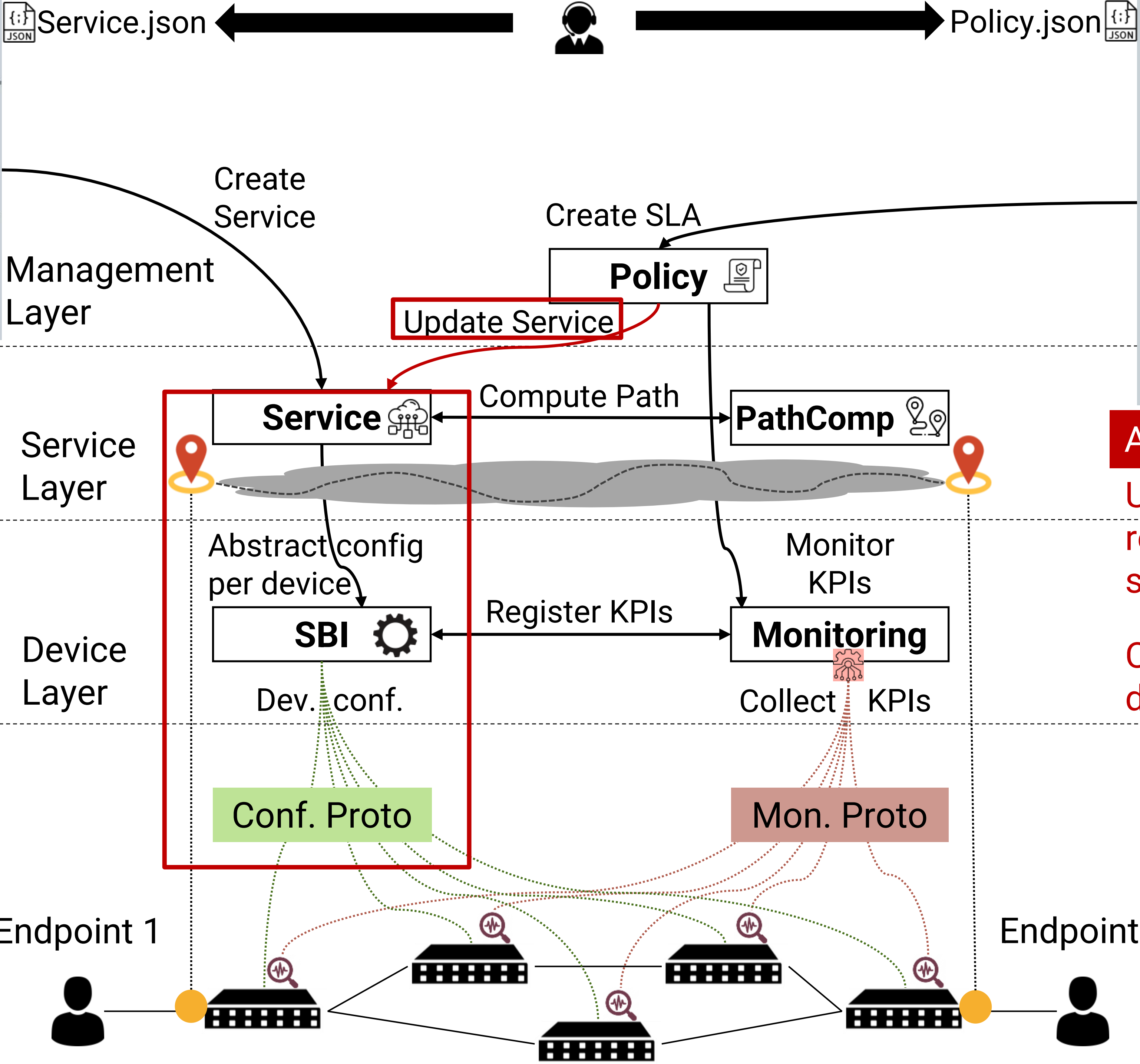
```
{ ⊟
  "serviceId":{ ⊟
    "context_id":{ ⊞ },
    "service_uuid":{ ⊞ }
  },
  "policyRuleBasic":{ ⊟
    "priority":0,
    "policyRuleId":{ ⊞ },
    "booleanOperator":"BOOLEAN_OR",
    "actionList":[ ⊞ ],
    "conditionList":[ ⊞ ]
  }
}
```

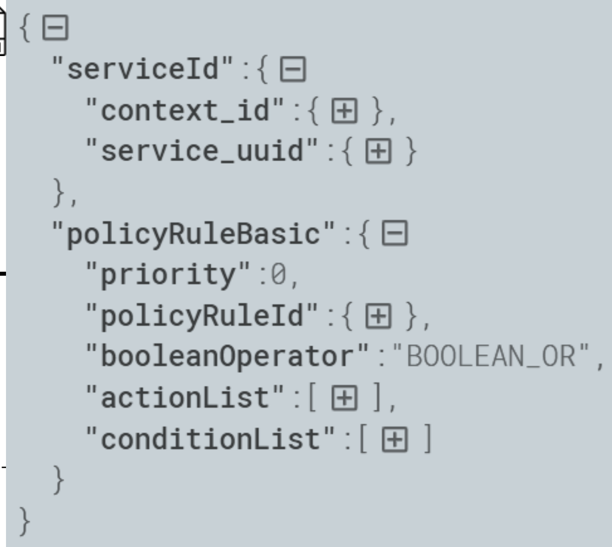
Action

Update service by requesting e.g., service path re-computation



```
{ ⊟
  "service_id":{ ⊟
    "service_uuid":{ ⊟
      "uuid":"svc:SW1/3==SW4/3"
    },
    "context_id":{ ⊞ }
  },
  "service_type":"L2NM",
  "service_endpoint_ids":[ ⊞ ],
  "service_constraints":[ ⊞ ],
  "service_config":{ ⊟
    "config_rules":[ ⊞ ]
  }
}
```





```
{ ⊟
  "serviceId":{ ⊟
    "context_id":{ ⊞ },
    "service_uuid":{ ⊞ }
  },
  "policyRuleBasic":{ ⊟
    "priority":0,
    "policyRuleId":{ ⊞ },
    "booleanOperator":"BOOLEAN_OR",
    "actionList":[ ⊞ ],
    "conditionList":[ ⊞ ]
  }
}
```

TeraFlow SDN by ETSI

Action

Update service by requesting e.g., service path re-computation

Creates cascade effect down to device-level

● Service Endpoint

Monitoring Agent

Monitoring Collector

Summary

Summary

Device-level microservices

- Handle the vast heterogeneity of the data plane
- Create device-specific and protocol-specific config

Summary

Device-level microservices

- Handle the vast heterogeneity of the data plane
- Create device-specific and protocol-specific config

Service-level microservices

- Ensure end-to-end connectivity between endpoints
- Take us from devices to networks
- Offer path computation as a service

Summary

Device-level microservices

- Handle the vast heterogeneity of the data plane
- Create device-specific and protocol-specific config

Service-level microservices

- Ensure end-to-end connectivity between endpoints
- Take us from devices to networks
- Offer path computation as a service

Management-level microservices

- With the support of Monitoring, closes a loop among:
  - Devices, services, and operators' objectives (policies or SLAs)
  - Auto-update service upon network state changes

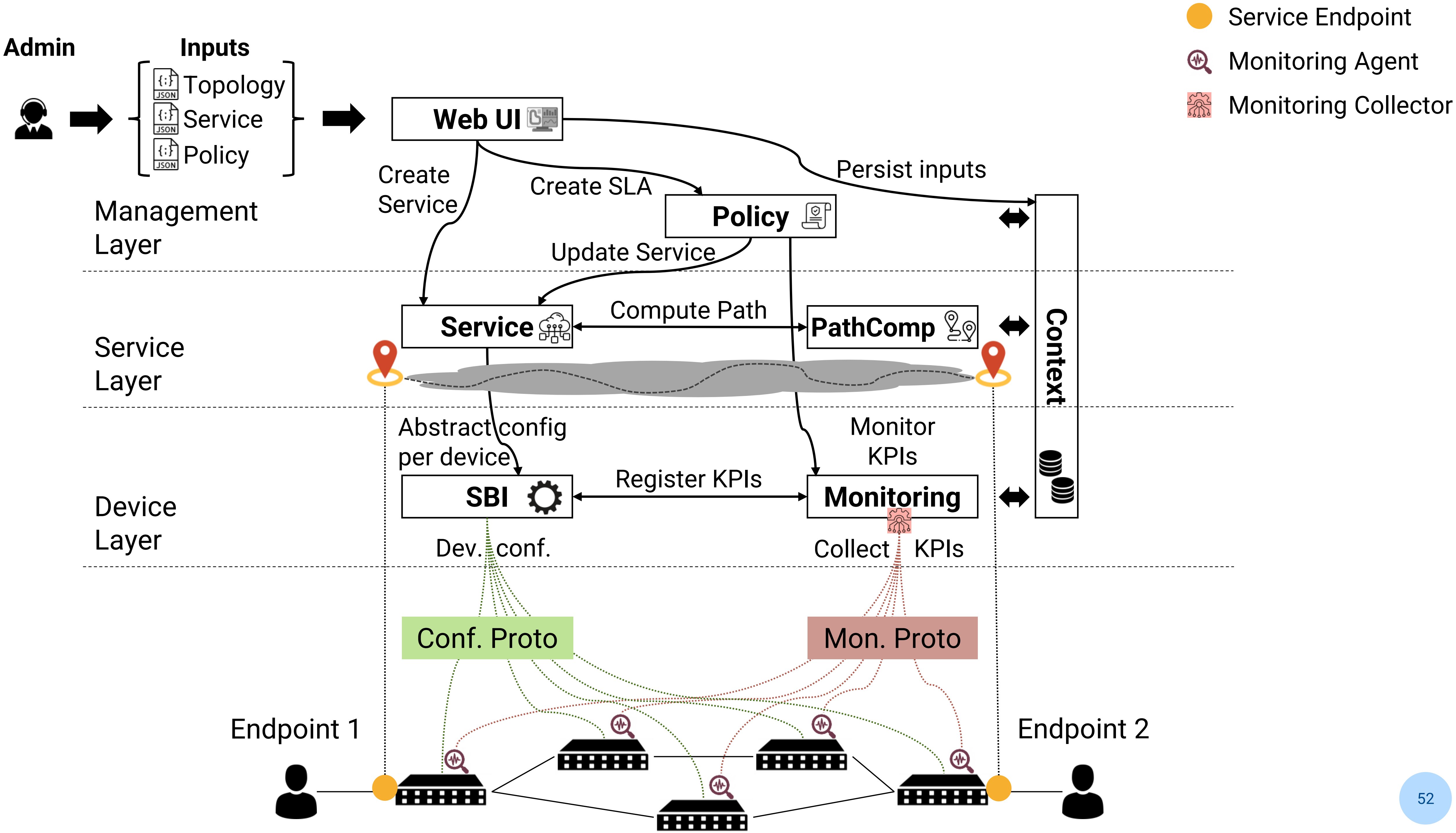
TeraFlow SDN

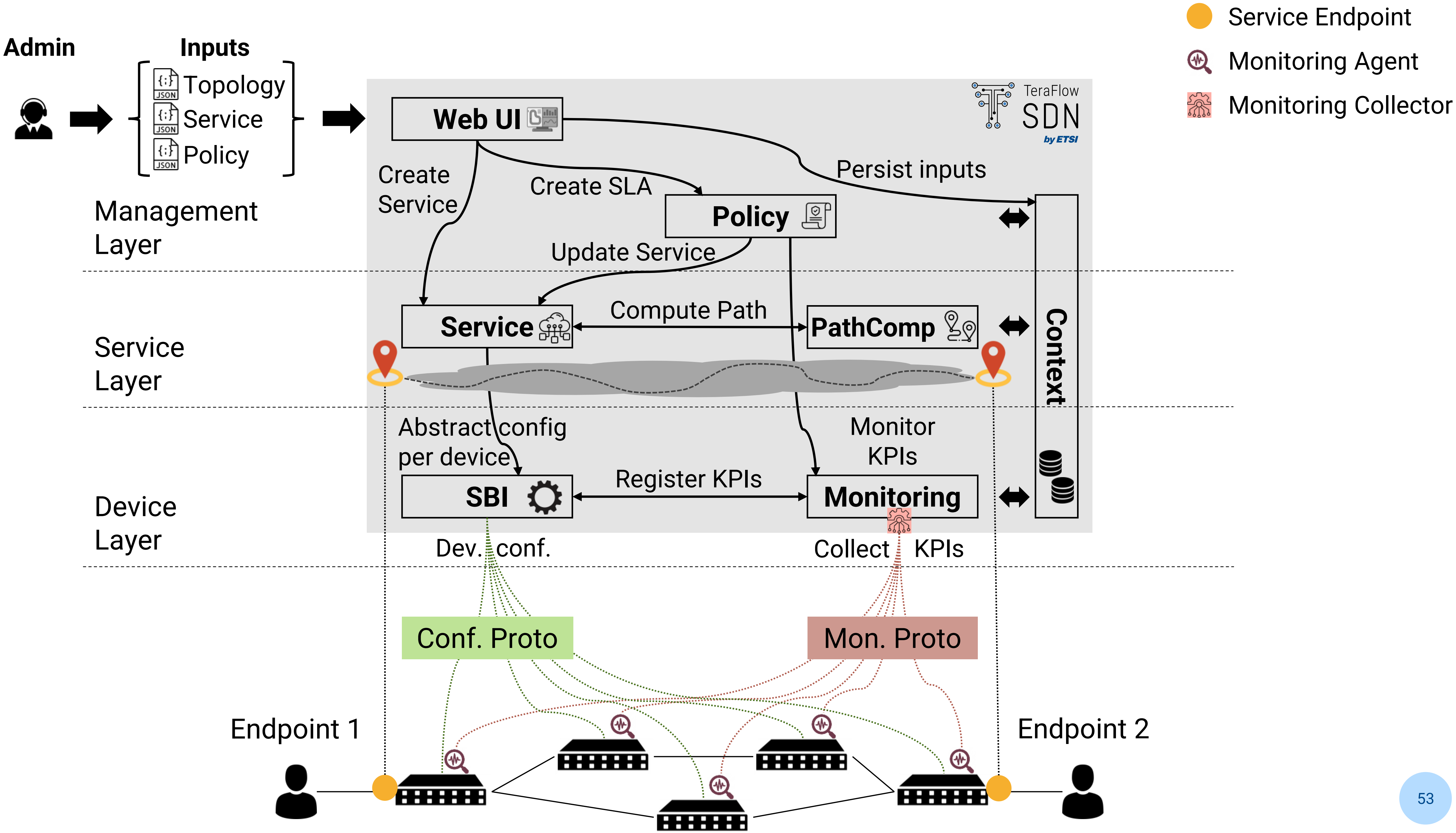
by ETSI

Thank you!

TFSsupport@etsi.org

Back-Up


Service Endpoint
Monitoring Agent
Monitoring Collector
Admin
Inputs
Topology
Service
Policy
JSON
Web UI
Create Service
Create SLA
Persist inputs
Policy
Update Service
Management Layer
Service
Compute Path
PathComp
Context
Service Layer
Abstract config per device
Monitor KPIs
SBI
Register KPIs
Monitoring
Device Layer
Dev. conf.
Collect KPIs
Conf. Proto
Mon. Proto
Endpoint 1
Endpoint 2



● Service Endpoint

Monitoring Agent

Monitoring Collector

Admin

Inputs

Topology

Service

Policy

TeraFlow SDN by ETSI

Web UI

Create Service

Create SLA

Persist inputs

Policy

Update Service

Management Layer

Service

Compute Path

PathComp

Context

Service Layer

Abstract config per device

Monitor KPIs

SBI

Register KPIs

Monitoring

Device Layer

Dev. conf.

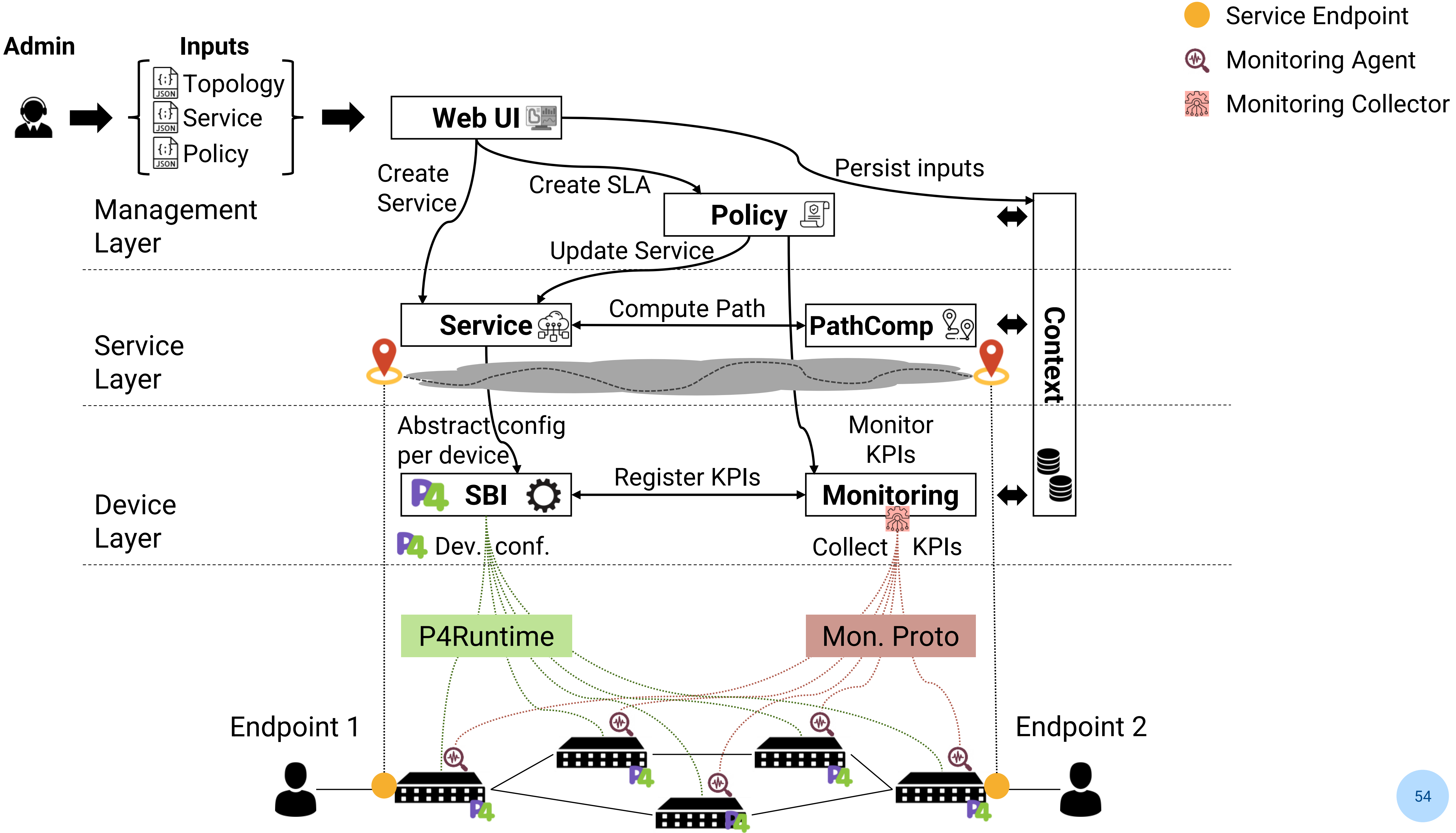
Collect KPIs

Conf. Proto

Mon. Proto

Endpoint 1

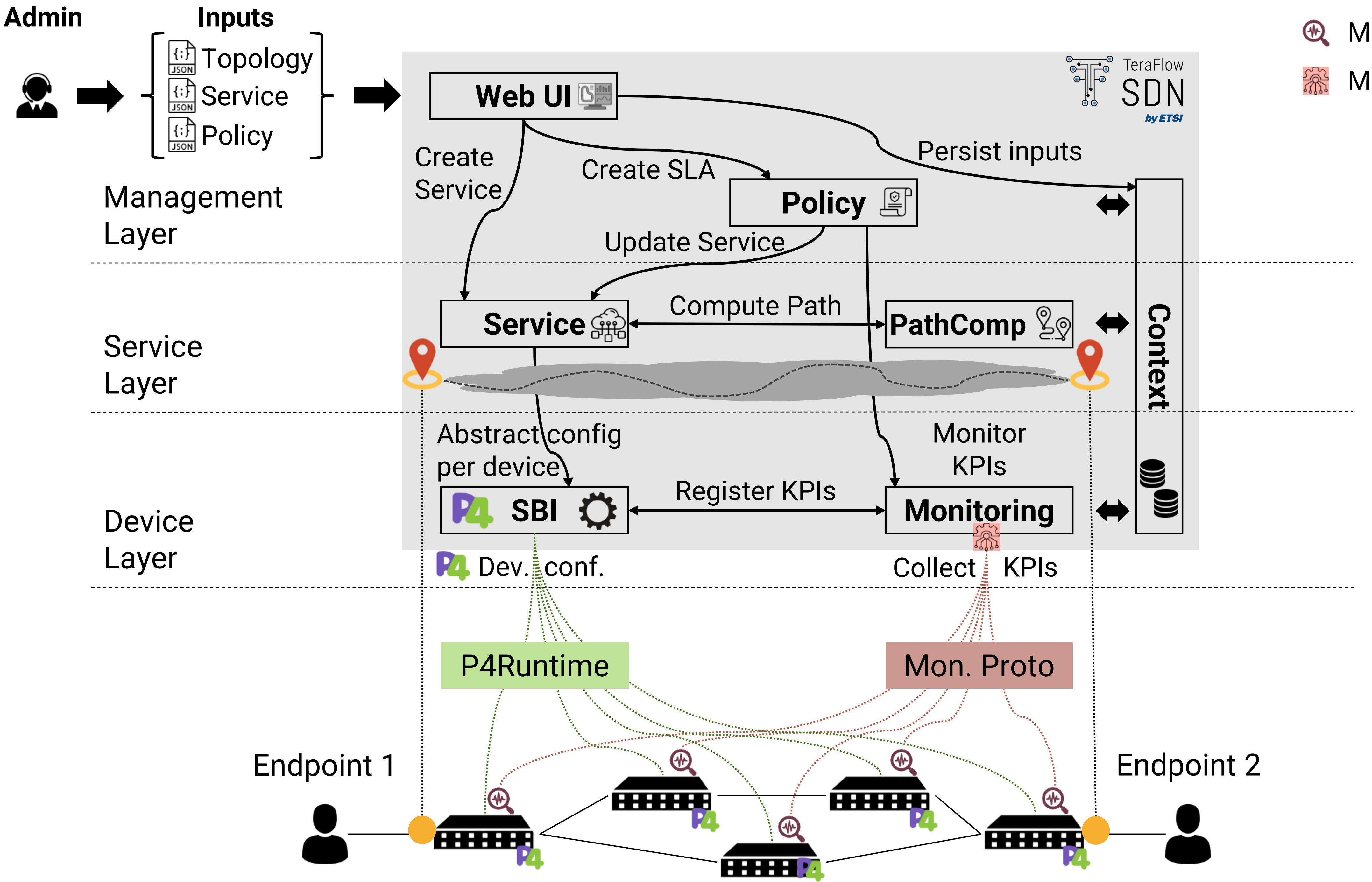
Endpoint 2



● Service Endpoint

Monitoring Agent

Monitoring Collector



Admin

Inputs

Topology

Service

Policy

TeraFlow SDN by ETSI

Management Layer

Service Layer

Device Layer

Web UI

Create Service

Create SLA

Persist inputs

Policy

Update Service

Service

Compute Path

PathComp

Context

Abstract config per device

Monitor KPIs

SBI

Register KPIs

Monitoring

Dev. conf.

Collect KPIs

P4Runtime

Mon. Proto

Endpoint 1

Endpoint 2