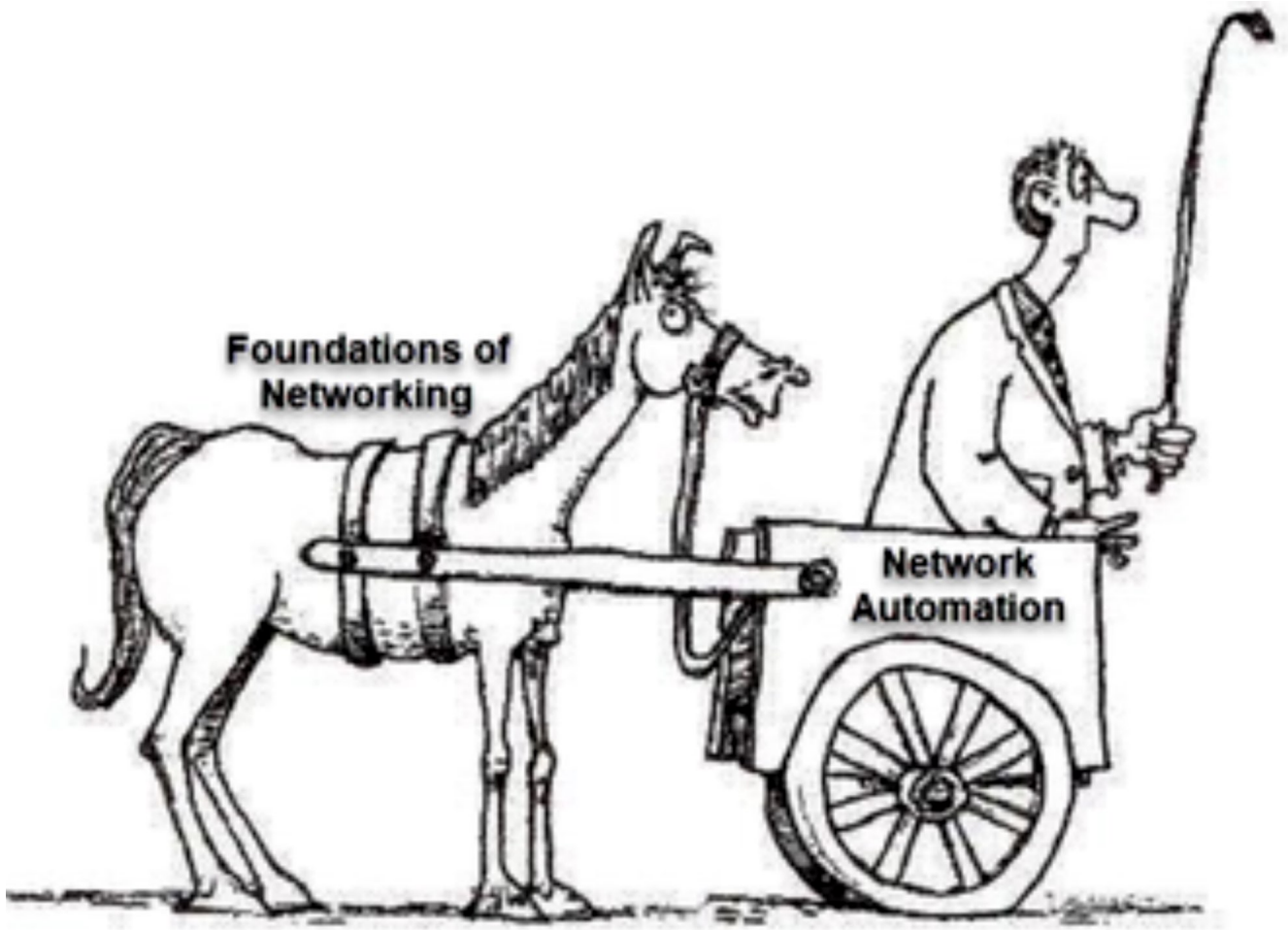
Network Automation Evolution

IETF #105

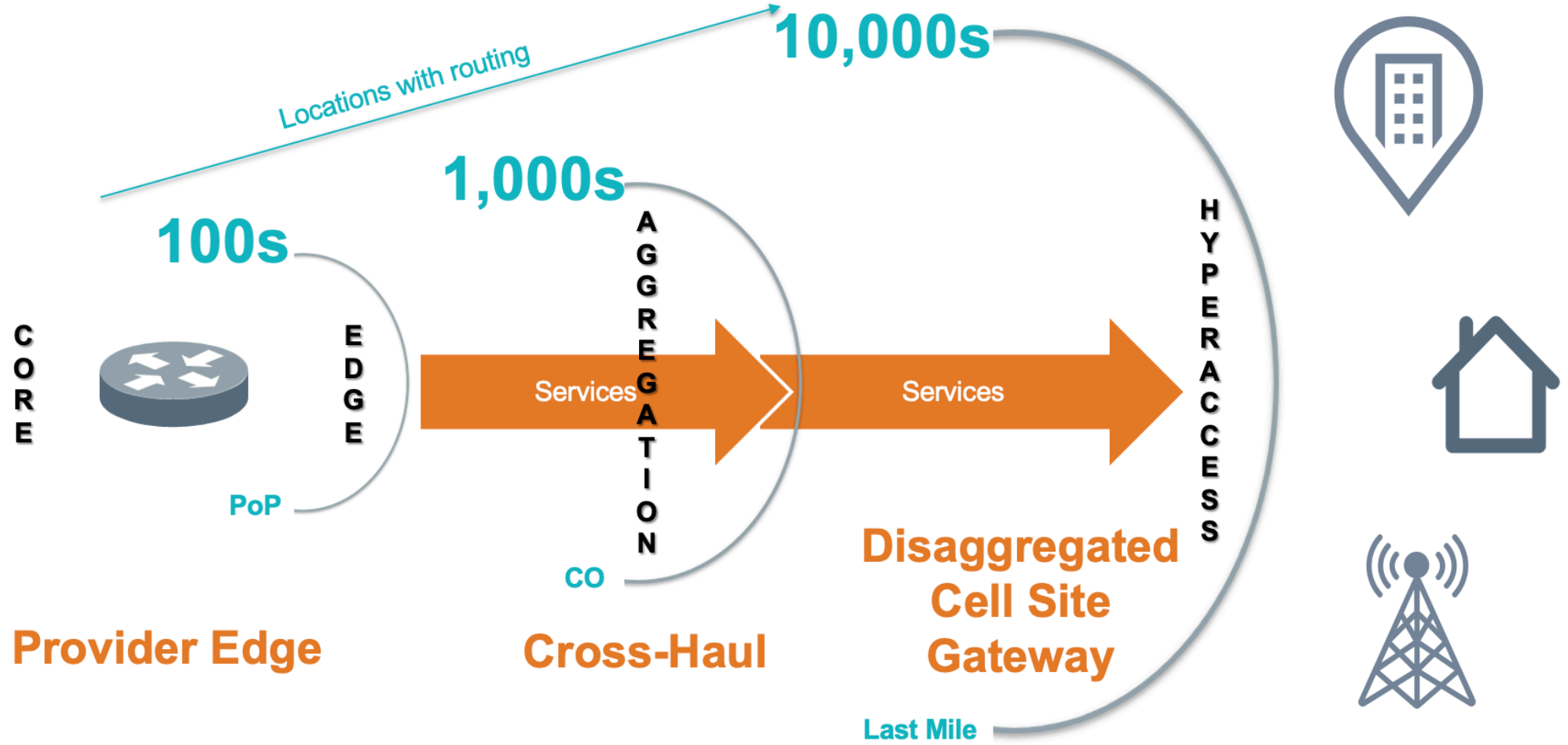
Montreal, July 2019

Josep-Lluis Ferrer, Raymond Cheh, Dean Bogdanovic
Volta Networks



Source: Twitter

New Service Edge Changes Routing



Traditional Network Management

- Networks are simply considered as collection of switches and routers.
- This practice leads to more complexity, due to increased number of systems that must be managed directly. Non-scalable.
- Lack of programmability, forced to think in terms of device configuration.
- Lack of agility on delivering new features. Locked to equipment provider release cycles.
- Tools and protocols: CLI, SSH, SNMP, custom tools.

- **Network automation** is a methodology in which software automatically configures, provisions, manages and tests network devices
- *(with minimal or no human intervention).*
- It can be achieved by means of **programmatic** capabilities, i.e., via **APIs** which define the supported remote calls.

API Based Management Approach

- All network elements could be configured using the same tools and abstractions
- Software based workflow to increase agility
- Network operator could focus on build new services
- Network engineers could build custom services with common set of tools
- Tools and protocols: YANG, NETCONF, Openconfig, gRPC, vendor SDKs, open source libraries

API Definition Requirements

- **Data model**: Define the data consumed by the methods (e.g. YANG)
- **Operations**: Define the operations that can be performed via API.
- **Serialization**: The encoding, how data is sent over the wire (examples JSON, XML, protobuf)
- **Transport**: The underlaying protocol to consume the API calls. E.g., HTTP, HTTP/2, QUIC, SSH…

Data Model

- YANG (IETF) adopted as main data-modeling language for networking devices, providing both configuration and operational state (including statistics)
- Defines data hierarchy as tree structure
- Specifies data types, restrictions (read, read+write), valid values, defaults
- Can be converted to any encoding format: JSON, XML
- **Open** models (vendor neutral): IETF, Openconfig
- Vendor models

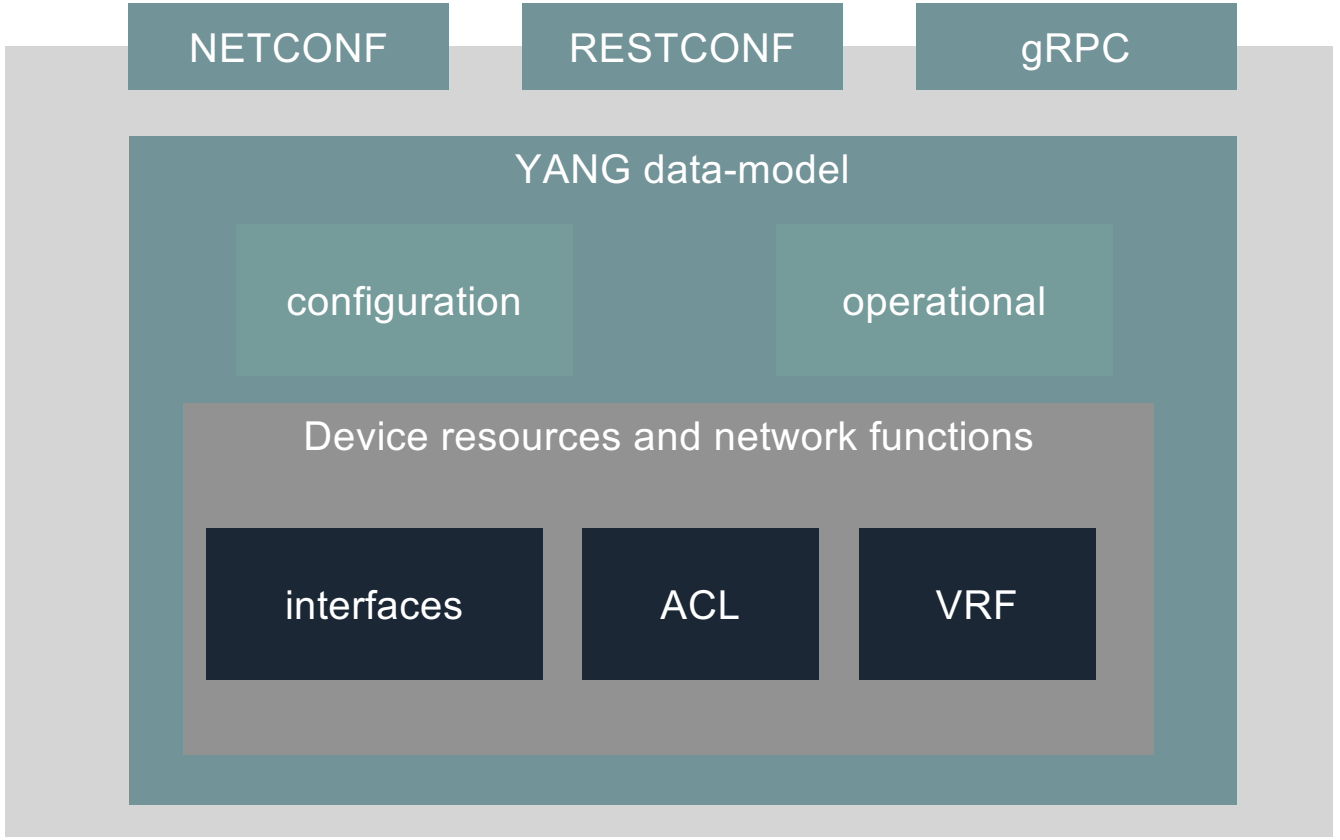
Transport Protocols

- **NETCONF (SSH)**
  - RPCs (XML/JSON): GET-CONFIG, EDIT-CONFIG, COMMIT,…
- **RESTCONF (HTTP/S)**
  - RPCs (XML/JSON): GET, POST, DELETE, PUT
- **gRPC (HTTP/2)**
  - RPC: Req/Rsp, streaming, bidirectional, …
  - —> De-facto standard for telemetry

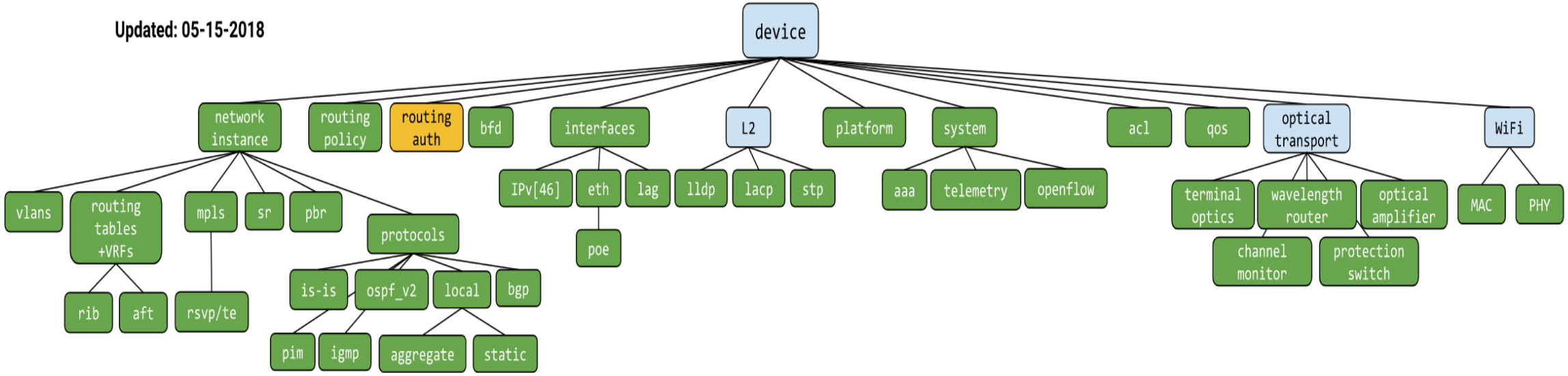
HTTP/2

Request/Response multiplexing
Bidirectional streams
Binary framing
Streams priorization

Network Device Configuration And Management APIs

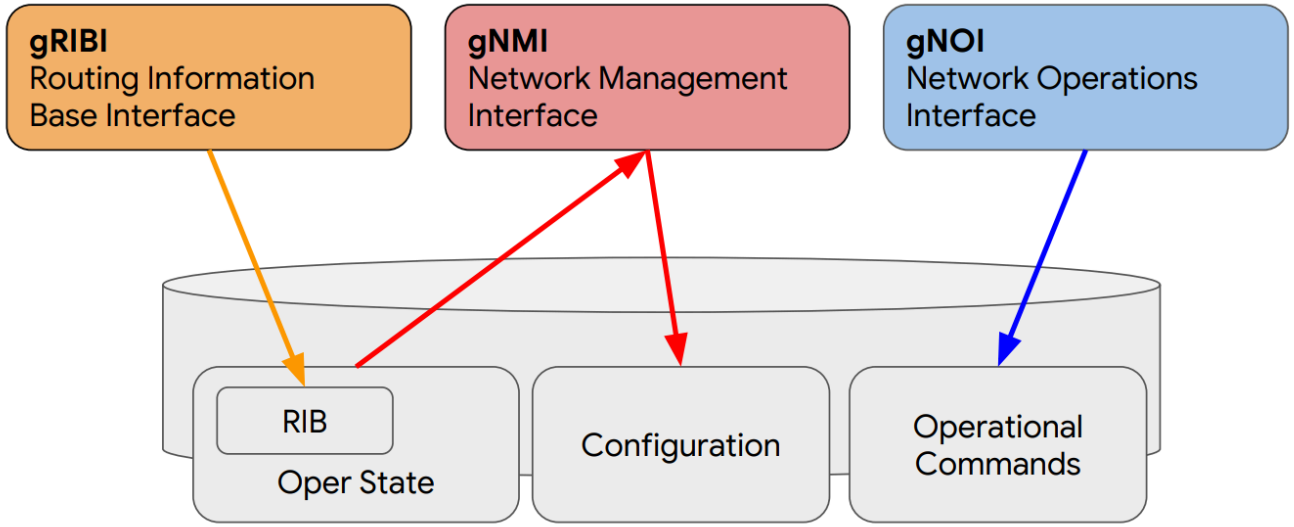


OpenConfig: Device Models*



*http://www.openconfig.net/projects/models/

OpenConfig: gRPC Interfaces



gNMI Figure source: https://datatracker.ietf.org/meeting/101/materials/slides-101-rtgwg-sessa-grpc-services-on-network-devices-00

gNMI

```
service gNMI {

  rpc Capabilities(CapabilityRequest) returns (CapabilityResponse);

  rpc Get(GetRequest) returns (GetResponse);

  rpc Set(SetRequest) returns (SetResponse);

  rpc Subscribe(stream SubscribeRequest) returns (stream
SubscribeResponse);

}
```

gNMI Telemetry

```
message Subscription {
  Path path = 1;
  SubscriptionMode mode = 2;
  uint64 sample_interval = 3;
  bool suppress_redundant = 4;
  uint64 heartbeat_interval = 5;
}

message Path {
  repeated string element = 1
[deprecated=true];
  string origin = 2;
  repeated PathElem elem = 3;
  string target = 4;
}

message PathElem {
  string name = 1;
  map<string, string> key = 2;
}
```

- SubscribeRequest message allows multiple subscriptions via SubscriptionList message.
- Each SubscriptionList includes multiple Subscription messages
- Modes
  - STREAM: Sends value on change
  - ONCE: Only sends 1 update
  - POLL: Actively poll for the value
- Path and PathElem represent serialization of XPATHs telemetry clients can be subscribed —> XPATH is text based
- gNMI encoding (TypedValue):
  - JSON
  - BYTES
  - PROTO
  - ASCII
  - JSON_IETF
  - Native (int, bool,…)

Adding Value To The Networks: Do We Have The Right Tools?

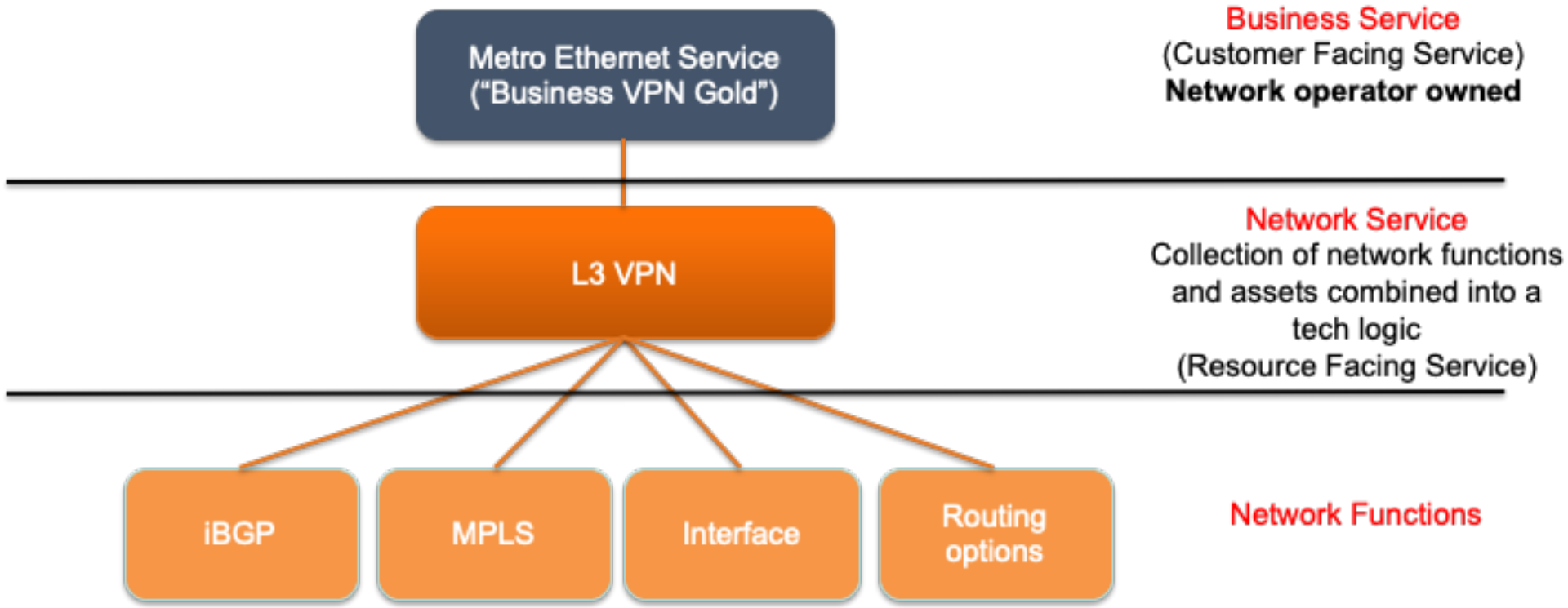
Network Service is a collection of network functions and device resources combined into a business and/or technology logic distributed among different network elements.

Network Function describes the configuration parameters of a specific device technology or feature and exposes via API (ACLs, routing protocols, policies,…)

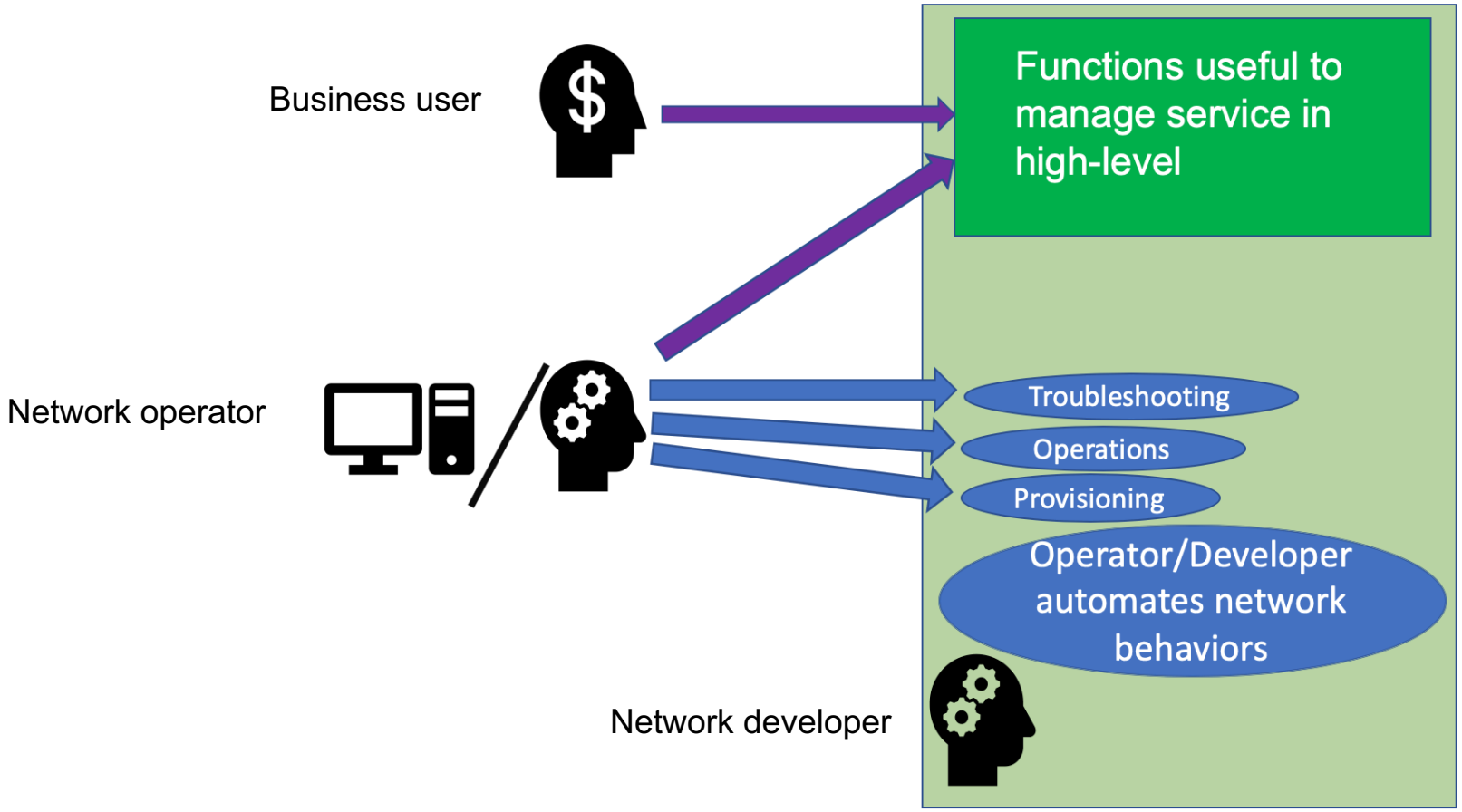
Services Models

- IETF also defines service level YANG models
  - L2VPN (RFC8466) and L3VPN (RFC8299)
- Openconfig only defines models at device level
- Network operators and architects still have to create their own tools to create and manage services (and create value!!)
- **Could we use any re-usable pattern to design and automate networking services?**

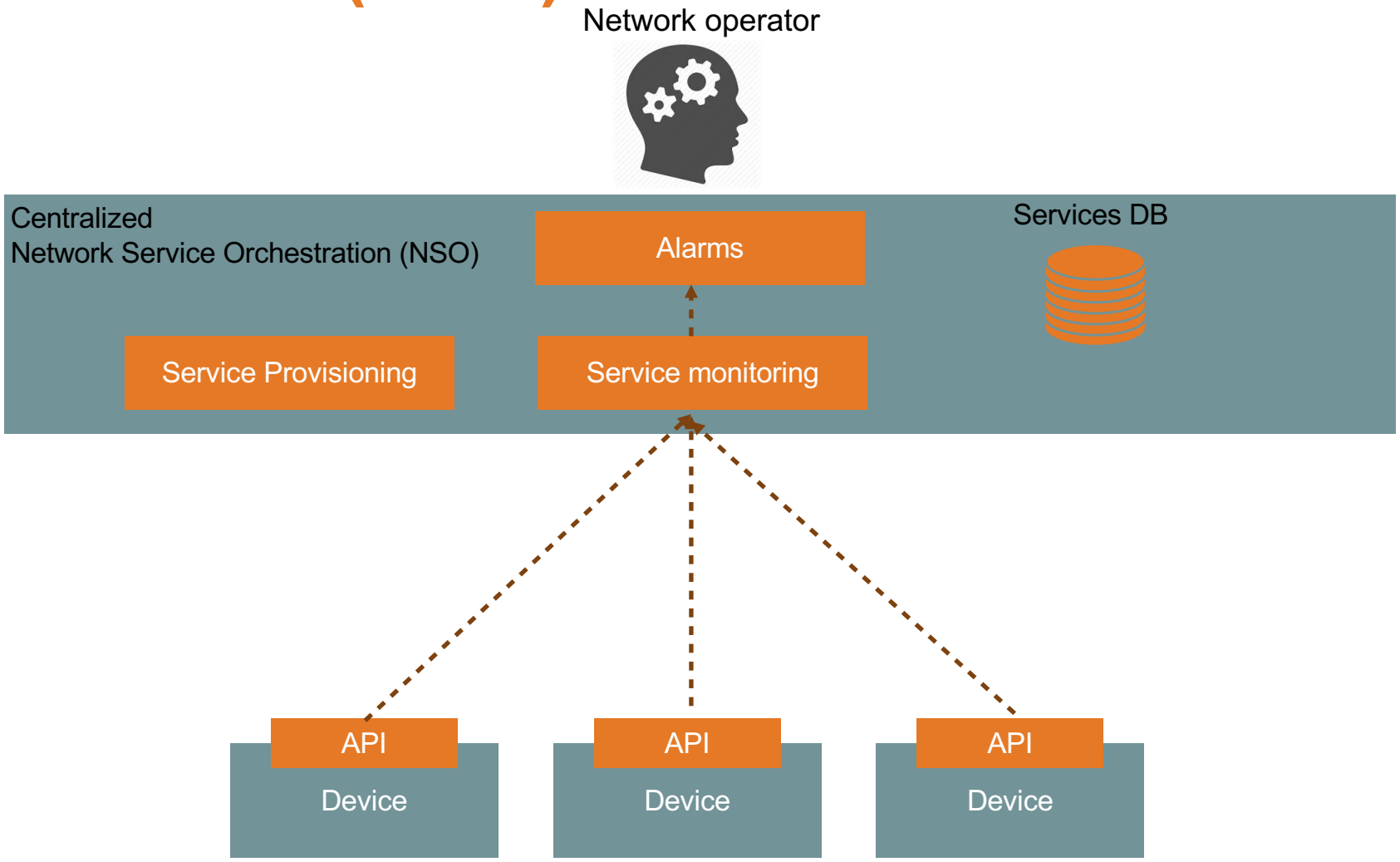
Network Service Example



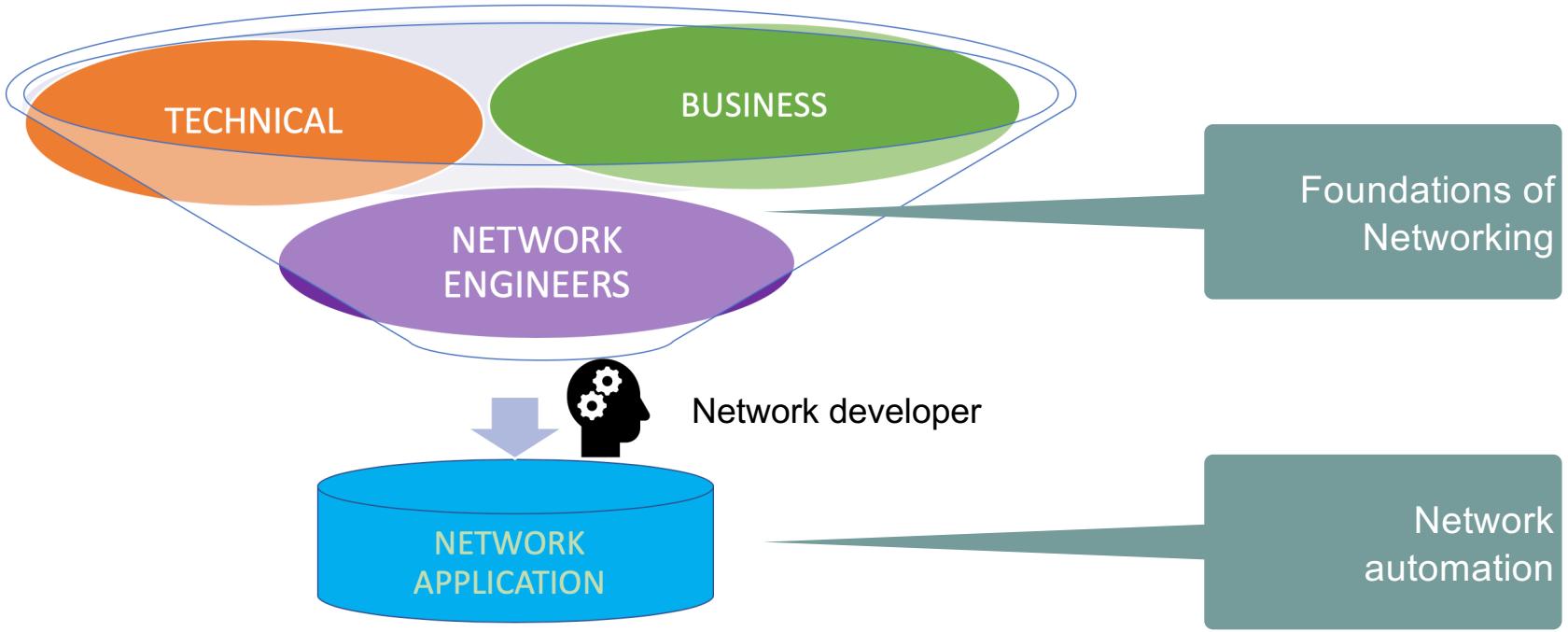
Network Service Automation



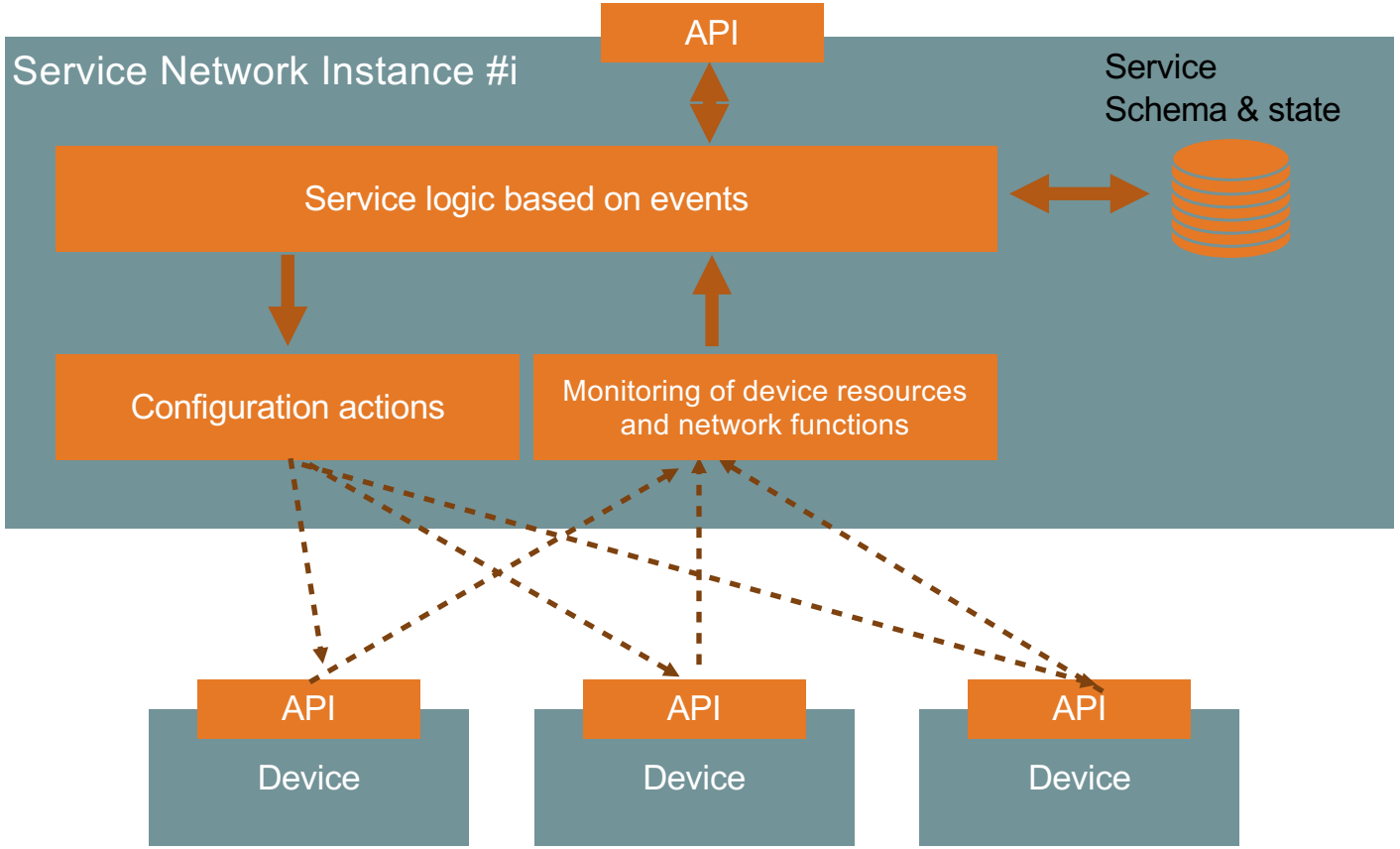
Traditional Approach: Network Service Orchestration (NSO)



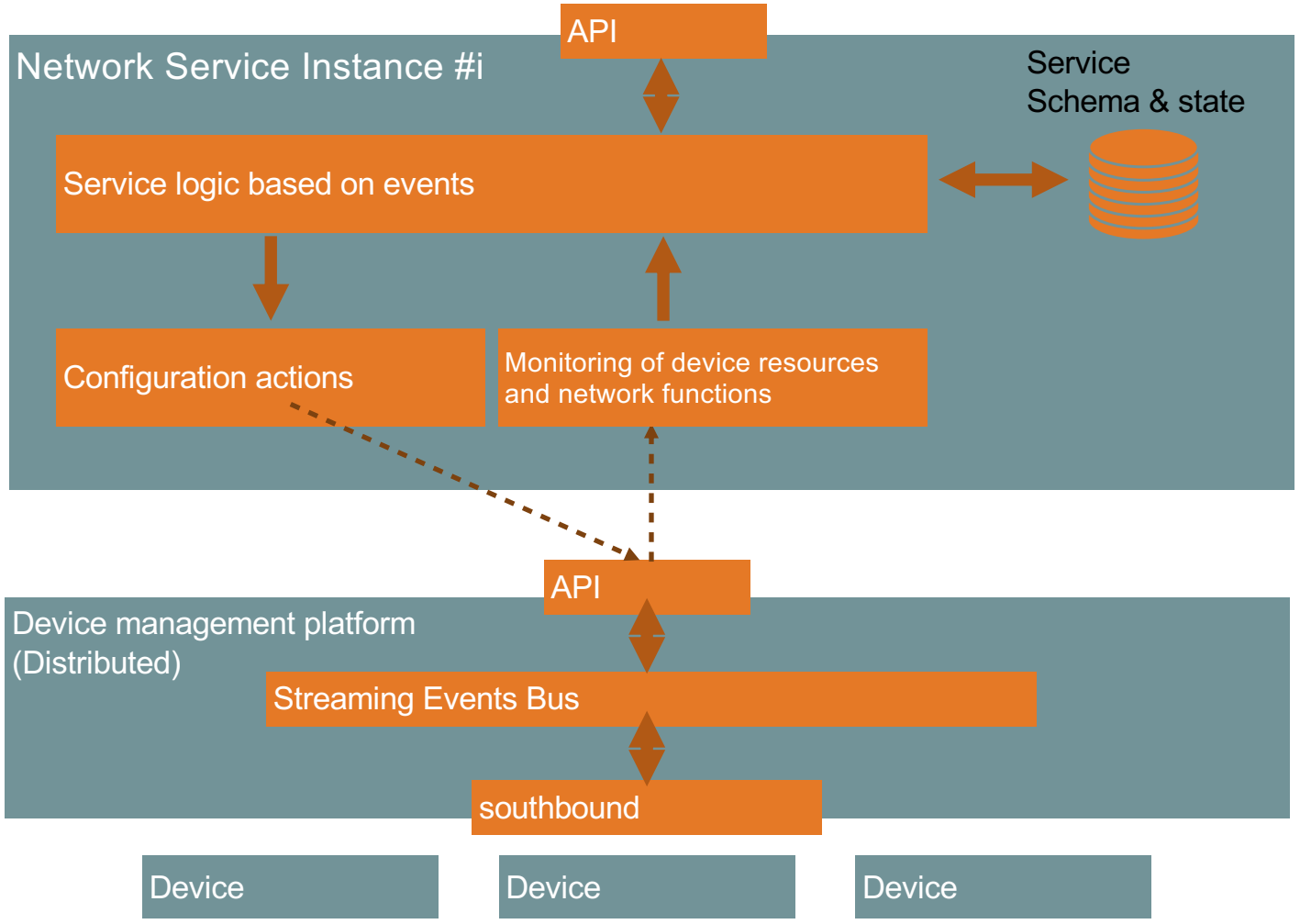
Network Service As SDN Application



Network Service As SDN network Application (II)



Network Service As SDN network Application (III)



Network Service As SDN network Application (IV)