27-06-2023

Milestone M7.3
Common NREN Network Service Product Models

Contractual Date:	30-06-2023
Actual Date:	27-06-2023
Grant Agreement No.:	101100680
Work Package:	WP7
Task Item:	Task 3
Nature of Milestone:	Supporting Document
Dissemination Level:	PU (Public)
Lead Partner:	SURF
Document ID:	GN5-1-23-432b23
Authors:	Hans Trompert (SURF)

Abstract

This document describes a set of network service product models that are common to the NREN community and can be used in combination with the Workflow Orchestrator. Models for a node, port, L2 point-to-point service, L2 VPN service, and IP static service, are presented, along with descriptions of how the model building blocks can be extended and re-used.

Table of Contents

Table of Figures

Executive Summary

A growing number of NRENs are interested in automating and orchestrating their network portfolio. Of the many commercial and open-source tools available, the NREN community's interest appears to be focused on Ansible and NSO for the automation part and on Workflow Orchestrator – an open-source framework originally written by SURF and now used by SURF, ESnet and GÉANT [M7.2] – for the orchestration part. This document describes a set of network service products that are common to this community and can be used in combination with the Workflow Orchestrator.

A common way of modelling products is to split the models into a customer-facing part that contains all the attributes that are significant to the customer and a resource-facing part that extends that set of attributes with all the attributes that are needed to actually deploy a service on the network. A set of network service product models and their attributes, for a node, port, L2 point-to-point service, L2 VPN service, and IP static service, is presented. It shows model building blocks that can be re-used and extended to the specific needs of the environment they are used in, as well as how these can serve as a basis for new constellations of product models.

A graph of building block instances is stored in the Workflow Orchestrator database by adding references from one model building block to another. This graph reflects the relations between the snippets of network node configuration that are deployed to the network. The graph is automatically added to when new services are deployed, allowing easy and intuitive navigation through all configuration data.

1 Introduction

Growing numbers of NRENs are interested in automating and orchestrating their network portfolio. However, individual NRENs may be at different levels of engagement, ranging from interested but with no concrete plans as yet, to fully automated and orchestrated. Of the many commercial and open-source tools that can be used, the NREN community's interest appears to be focused on Ansible and NSO for the automation part and on Workflow Orchestrator – an open-source framework originally written by SURF and now used by SURF, ESnet and GÉANT [M7.2] – for the orchestration part. This document describes a set of network service products that are common to this community and can be used in combination with the Workflow Orchestrator.

1.1 Standards

There are many standards describing how network service products and their attributes can be modelled. Most of these are very detailed as they try to cover as many use cases as possible, which can prove overwhelming. This document aims to do the opposite and only model the bare minimum. This makes it easier to see the relationship between the network service models, and how each model can be extended with attributes that are specific to the organisation that uses them.

A common way of modelling products is to split the models into a customer-facing part that contains all the attributes that are significant to the customer, and a resource-facing part that extends that set of attributes with all the attributes that are needed to actually deploy a service on the network. In this document we assume that such a separation is being used, where the customer-facing part lives in the Workflow Orchestrator and the resource-facing part lives in a provisioning system such as NSO or Ansible.

1.2 Modelling

There are several ways in which network service products can be modelled and split up into logical parts (product blocks). This may for example depend on the requirements from the stakeholders, the environment such products are used in, and/or personal taste. It is important to highlight that the modelling process aims to identify the main attributes necessary to describe a product and the relations between product blocks – it is not prescriptive in terms of design and implementation of a specific product. This approach helps to decouple the two topics and keep them in separate functional domains.

Looking at different sets of network service product models we do see that they share a set of core attributes, regardless of which product or product block they belong to. This is not surprising, because the key attributes needed to actually provision a network service on the network are the same for all of these. This document describes a set of network service product models and their attributes that can either be extended to meet the specific needs of the environment they will be used in, or can serve as a basis for a new constellation of product models. There is no intention to supply a complete set of products that covers all possible NREN network services, but it should be enough to help inspire thinking about (network) product modelling in your organisation.

1.3 Context

The models described below assume an Ethernet network that consists of nodes where each node has physical ports. Network services have endpoints that connect to ports. The attributes that are specific to an endpoint are modelled as a service attach point. Examples of such attributes are the layer two label(s) used on that port or a point-to-point IP address. An inventory management system (IMS) is used to keep track of everything that is being deployed, and a network resource manager (NRM), such as NSO or Ansible, is used to provision the services on the network. All IP addresses and prefixes are stored in an IP address management (IPAM) tool.

1.4 Workflow Orchestrator Terminology

The data and business rules of the products and product blocks are modelled in Workflow Orchestrator domain models. A product is a collection of one or more product blocks, and zero or more fixed inputs. Fixed inputs are customer-facing attributes that cannot be changed at will by a customer because they are constrained in some way, for example by a physical constraint such as the speed of a port or a financial constraint such as the maximum capacity of a service. Product blocks are collections of resource types (customer-facing attributes) that together describe a set of attributes that can be repeated one or more times within a product and can optionally point to other product blocks. A product block is a logical collection of resource types that taken together make reusable instances. They can be referenced many times from within other products and make it possible to build a logical topology of the network within the orchestrator database. A subscription is a product instantiation for a specific customer. See the Workflow Orchestrator documentation [Orchestrator Core] for more details.

2 Network Service Product Models

2.1 Node

The administration handoff in IMS will be different for every organisation. For this example, it is assumed that all administration that comes with the physical installation and first-time configuration of the network node in IMS is done manually by a NOC engineer. This makes the node product rather simple. The only product block that is defined holds pointers to all related information that is stored in the operations support systems (OSS). This includes of course a pointer to the information in IMS, and after the service has been deployed on the network, another pointer to the related information in the NRM. To keep track of all IP addresses and prefixes used across the network service product, the pointers to the IPv4 and IPv6 loopback addresses on the node are also stored.

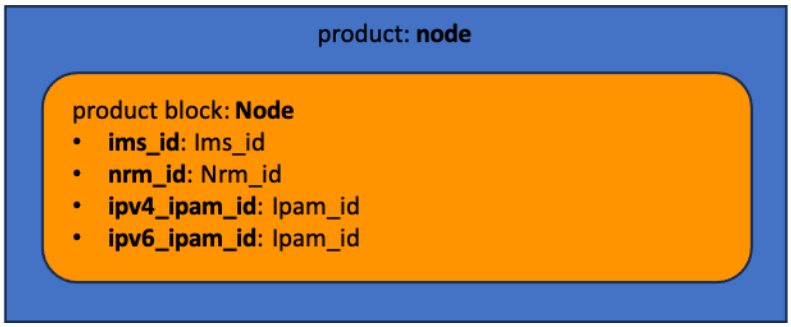

Figure 1: Node product model

- **ims_id**: ID of the node in the inventory management system
- **nrm_i**d: ID of the node in the network resource manager
- **ipv4_ipam_i**d: ID of the node's iPv4 loopback address in IPAM
- **ipv6_ipam_i**d: ID of the node's iPv6 loopback address in IPAM

2.2 Port

Once a NOC engineer has physically installed a port in a node and added some basic administration to IMS, the port is marked as available and can be further configured through the port product. To distinguish between ports with different speeds (1Gbit/s, 10Gbit/s, etcetera), the fixed input *speed* is used, which also allows filtering available ports of the right speed. Besides pointers to the administration of the port in IMS and the NRM, configuration options including 802.1Q, Ethernet auto negotiation, and the use of LLDP are registered, as well as a reference to the Node the port is installed in.

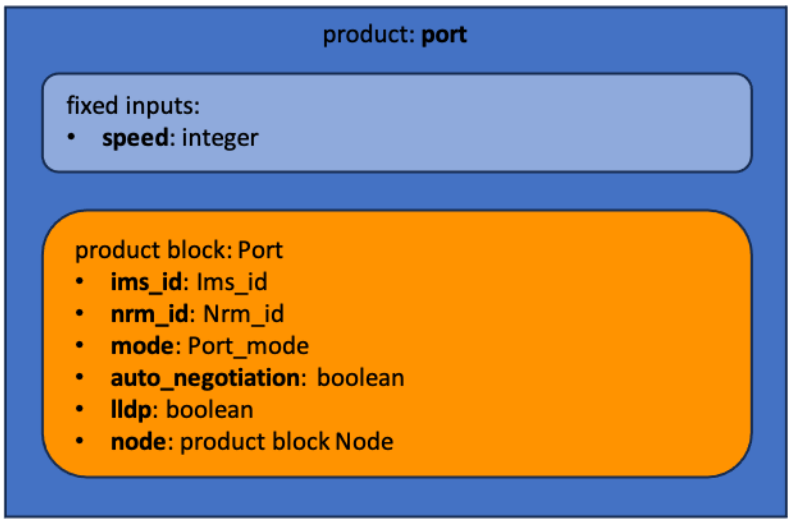

Figure 2: Port product model

- **speed**: the speed of the physical interface on the node in Mbit/s
- **ims_id**: ID of the node in the inventory management system
- **nrm_id**: ID of the node in the network resource manager
- **mode**: the port is either untagged, tagged or a link member in an aggregate
- **auto_negotiation**: enable Ethernet auto negotiation
- **lldp**: enable the link layer discovery protocol
- **node**: link to the Node product block the port is residing on

2.3 L2 Point-to-Point

The Layer 2 point-to-point service is modelled using two product blocks. The *l2_point_to_point* product block holds the pointers to IMS and the NRM, the speed of the circuit, and whether the speed policer is enabled or not, as well as pointers to the two service attach points. The latter are modelled with the *L2_service_attach_point* product block and keep track of the port associated with that endpoint and, in the case where 802.1Q has to be enabled, the VLAN range used. The service can either be deployed protected or unprotected in the service provider network. This is administered with the fixed input *protection_type*.

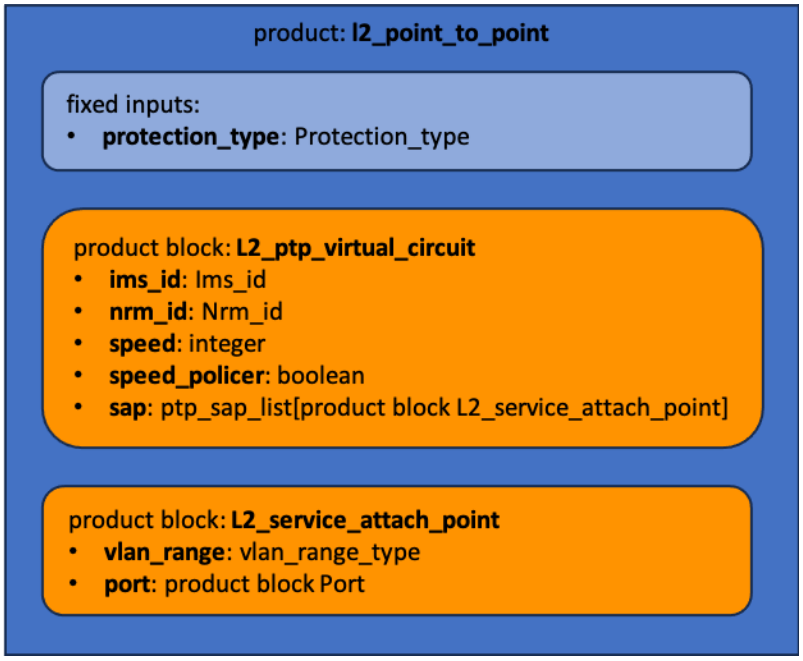

Figure 3: L2 point-to-point product model

- **protection_type**: this service is either unprotected or protected
- **ims_id**: ID of the node in the inventory management system
- **nrm_id**: ID of the node in the network resource manager
- **speed**: the speed of the point-to-point service in Mbit/s
- **speed_policer**: enable the speed policer for this service
- **sap**: a constrained list of exactly two Layer2 service attach points
- **vlan_range**: range of Layer 2 labels to be used on this endpoint of the service
- **port**: link to the Port product block this service endpoint connects to

2.4 L2 VPN

The Layer 2 VPN service is much like the Layer 2 point-to-point service, which makes it possible to reuse existing product blocks, with a few differences such as the absence of fixed inputs. The *L2_vpn_virtual_circuit* product block inherits from the *L2_ptp_virtual_circuit* product block, and adds attributes to (dis)allow VLAN retagging and control over the BUM filter. And because a VPN can have one or more endpoints, unlike a point-to-point that has exactly two endpoints, the list of service attach points is overridden to reflect this.

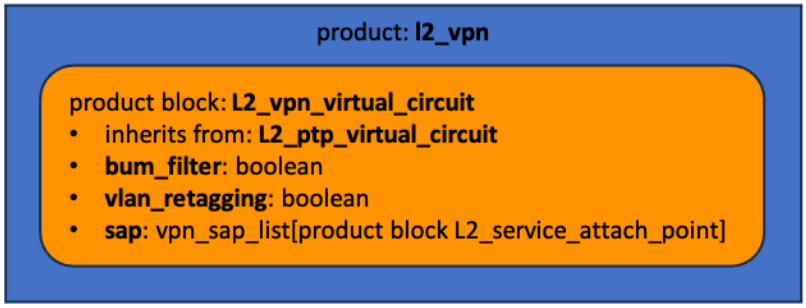

Figure 4: L2 VPN product model

bum_filter: enable broadcast, unknown unicast, and multicast (BUM) traffic filter

vlan_retagging: allow VLAN retagging on endpoints

sap: a constrained list of at least one Layer2 service attach point

2.5 IP static

The modelling of the IP static service is slightly more difficult. Luckily, we are again able to reuse existing product blocks and add or change attributes to meet our needs. First of all, a fixed input is used to distinguish between different types of IP services, in our case it is used to distinguish between static and BGP routing. The *Ip_static_virtual_circuit* product block reuses the *L2_ptp_virtual_circuit* product block and adds the ability to administer additional IP settings such as the use of multicast and whether a CERT filter is enabled or not. The list of service attach points is overridden, this time to reflect the fact that the IP static service only has one endpoint. The layer 3 service attach point extends the one at layer 2 and adds a list of customer prefixes, the IPv4/IPv6 MTU, and the IPv4/IPv6 point-to-point addresses used. For this example, we chose to bundle the IP settings in a separate product block to make it possible to be reused by other products, but we could also just have extended the *Ip_static_virtual_circuit* product block.

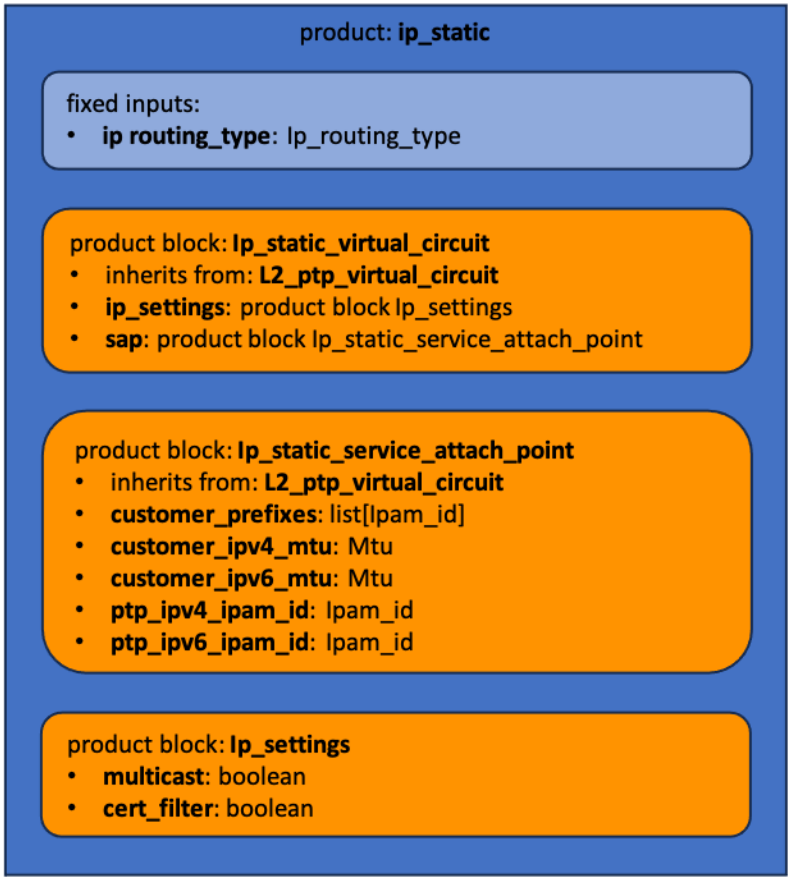

Figure 5: IP static product model

- **ip_routing_type**: either Static or BGP, for this product set to Static
- **customer_prefixes**: list of IPAM ID's of the customer IP prefixes
- **customer_ipv4_mtu**: the customer IPv4 maximum transmission unit
- **customer_ipv6_mtu**: the customer IPv6 maximum transmission unit
- **ptp_ipv4_ipam_id**: the IPAM id of the IPv4 point-to-point prefix
- **ptp_ipv6_ipam_id**: the IPAM id of the IPv6 point-to-point prefix
- **multicast**: enable multicast
- **cert_filter**: enable CERT filter

3 Product Block Instance Graph

A subscription for a specific customer for a product that is deployed on the network is stored in the Workflow Orchestrator database. The notion of subscription ownership allows for fine-grained control over which customer is allowed to change what attribute. By correctly adding references from one product block to another, a graph of product block instances is generated that accurately reflects the relations between the snippets of network node configuration that are deployed to the network. The graph is automatically added to when a new subscription is created, allowing easy and intuitive navigation through all configuration data. Once every network service is modelled and provisioned to the network through the Workflow Orchestrator, every line of network node configuration can be linked to the corresponding subscription that holds the configuration parameters.

The example below shows the product block instance graph for a L2 point-to-point and a L2 VPN service between three ports on three different nodes. The nodes are owned by the respective NREN's Network Operations Centre (NOC). University A has ports on nodes on two different locations, and uses a L2 point-to-point service to connect these locations. Research Institute B has one port of its own, and uses a L2 VPN service for their collaboration with the university. The business rules that describe the (optional) authorisation logic for connecting subscriptions from different customers to each other are coded in the Workflow Orchestrator workflows related to these products.

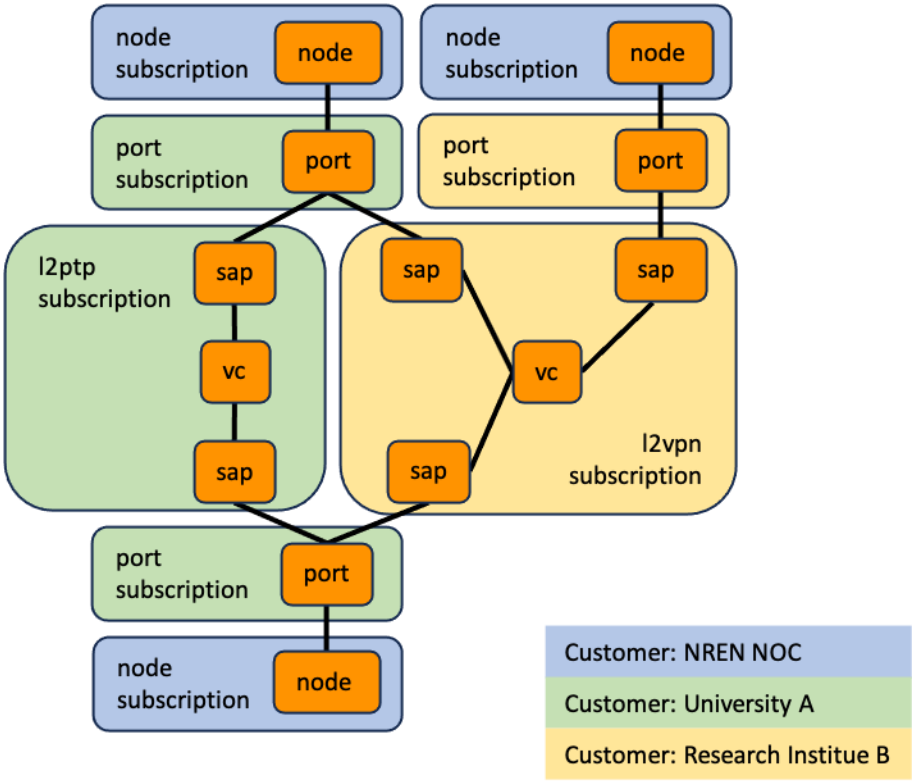

Figure 6: Product block instance graph example

References

[M7.2] *Orchestration Platform Architecture Review and Documentation* https://resources.geant.org/wp-content/uploads/2023/06/M7.2_Orchestration-Platform-Architecture-Review-and-Documentation.pdf

[Orchestrator Core] https://workfloworchestrator.org/orchestrator-core/

Glossary

802.1Q	Standard that supports VLANs on an IEEE 802.3 Ethernet network
BGP	Border Gateway Protocol
BUM	broadcast, unknown unicast, and multicast traffic
CERT	Computer Emergency Response Team
IMS	Inventory Management System
IP	Internet Protocol
IPAM	IP Address Management
L2	the data link layer, or layer 2, of the OSI model of computer networking
MTU	Maximum Transmission Unit
NOC	Network Operations Centre
NREN	National Research and Education Network
NRM	Network Resource Manager
NSO	Cisco Network Service Orchestrator
OSI	Open System Interconnection
OSS	Operations Support Systems
VLAN	Virtual Local Area Network
VPN	Virtual Private Network