A comparative analysis of community developed resource models for the optical transport network:
A series of findings from the field

D. King, Old Dog Consulting

**Abstract**—The TOUCAN and Metro-Haul projects are deploying smart optical metro infrastructure able to support traffic originating from heterogeneous radio access networks, addressing the anticipated capacity increase and its specific characteristics, e.g., mobility, low latency, low jitter etc. This infrastructure requires suitable optical transport models to support a wide variety of services and use cases with special emphasis on services from various industries vertical to the ICT. The comparative analysis discusses the suitability of various community developed resource models and their subsequent strengths and weaknesses when evaluated for the TOUCAN and Metro-Haul projects.

**Index Terms**—Optical Modeling, Transport Northbound Interface (T-NBI), Transport Application Programming Interface (T-API)

# 1 Introduction

Transport Operator (Operator) infrastructure is comprised of multiple technologies across network layers (traffic engineered optical and packet). Typically, these resources are separated into multiple transport domains, each using different network technologies, control interfaces and implementing forwarding policy with diverse goals.

Management, configuration, and troubleshooting processes rely extensively on human intervention, using Element Management Systems (EMS) and Network Management Systems (NMS) to translate high-level connectivity goals into individual device configurations, while service deployment is designed using whiteboards by the network planners [Ref 1]. Correspondingly, transport service delivery times for new connections may take many months, with significant portions of this time spent in the design and configuration phase of the deployment lifecycle.

The inflexibility and limited automation of transport network, led to the development of new control and management architectures and protocols. We often to refer to this technology as Software Defined Networking (SDN): centralized control, separation of control and forwarding, open Application Programming Interface (API), and automation.

Existing optical transport networks often have separation of data plane and control elements; therefore, these are not new concepts, however establishing an open and well-defined method for exposing transport capability via a Transport Application Programming Interface (T-API), is now critical.

Potential success of Transport SDN in commercial environments is largely dependent on the success in specifying, documenting and standardising open transport interfaces between the Transport Orchestrator (T-O), Transport Controller (T-C) (Northbound Interface – NBI) and between TCs (East-West Interface).

A common open interface to each boundary is pre-requisite for network operators to control multi-vendor and multi-domain networks and also enable service provisioning coordination/automation. This must be achieved by using standardised models, used together with an appropriate messaging protocol (interface).

Several popular optical and transport SDN architectures and interfaces are being developed, including:

1. Generic functional architecture of transport networks [1], developed by the ITU Telecommunication Standardization Sector (ITU-T);
2. T-API developed by Open Networking Foundation (ONF) Requirements [2] and Architecture [3];
3. Transport Northbound Interface Use Cases [3], Abstraction and Control of Traffic-Engineered Networks (ACTN) [4], Traffic Engineering (TE) Topology [5] and TE Tunnels [6] models defined by the Internet Engineering Task Force (IETF).

This document highlights the key components of control, interaction and naming of transport SDN functions, important use cases and requirements, and the type and scope of information that must be exchanged over the key interfaces.

# 2 Transport SDN

Transport network domains, including Optical Transport Network (OTN) and Wavelength Division Multiplexing (WDM) networks, are typically deployed based on a single vendor or technology platforms. They are often managed using proprietary interfaces to dedicated Element Management Systems (EMS), Network Management Systems (NMS) and increasingly Software Defined Network (SDN) controllers.

A well-defined open interface to each domain management system or controller is required for network operators to facilitate control automation and orchestrate end-to-end services across multi-domain networks. These functions may be enabled using standardized data models (e.g. YANG), and appropriate protocol (e.g., RESTCONF).

## 2.1 Transport Service Perspectives

The following examples provide different use case perspectives for commercial transport SDN deployments.

1. **End-to-End Service Management**

Automated service creation covering Layer-0 to Layer-3.

1. **Elastic Bandwidth Provisioning**

Creation of elastic services with automatic or “on demand” changes in bandwidth.

1. **Dynamic Datacenter Interconnections**

Automatic load dependent fast service creation.

1. **Transport as a Service (TaaS)**

Fully automate service requests including network planning and node configuration.

1. **Multi-layer Network Operation**

Multilayer optimized Layer-0 to Layer-3 networking with automatic setup and teardown.

1. **Vendor Agnostic Transport Networking**

Standardised transport SDN control interfaces for automated integration and deployment of services across multi-vendor equipment.

## 6.3 Transport SDN Architecture

The architecture of SDN is specified in the ONF SDN architecture document [3], which identifies core principles of SDN and applies them to architectural components and interfaces.



**Figure 1: ONF Transport SDN Architecture**

SDN is often used in greenfield Ethernet and packet networks. An important design goal for application of these SDN principles to transport networks, is to be based SDN for transport on standardized and open interfaces, to overcome the existing inter-operability limitations created by the lack of integration and interoperability for transport network devices.

Essentially, there is a clear need a well-defined T-NBI and corresponding resource models. Combined, they are crucial for transport service orchestration, since they enable control and monitoring of service connectivity and network resource utilization and definition of custom fault-management processes. Nonetheless, NBI standardization is limited.

Existing SDN control interfaces and mechanisms have been driven by multiple open source (de facto standards), and formal Standards Development Organizations (SDOs) have been slow to develop proposals. There are some initiatives to program directly the optical devices from the South Bound Interface (SBI) via YANG models. This initial work represents a firm basis for interoperable and open resource models and is complementary to the NBI definition, which will enable the integration of current deployments as well as a smooth migration of the transport network to the SDN paradigm.

## 2.3 Transport Service Orchestration

Orchestration is a hot topic of current industry conversation dealing with network evolution. However, formal definitions do not exist, it remains an area where the meaning and scope of orchestration is often only implied.

Current understandings of orchestration include “the idea of automatically selecting resources to satisfy client demands” [7], which also defines orchestration as “the ongoing selection and use of resources by a provider to satisfy client demands according to optimization criteria.” This definition is intended to compactly encompass all the necessary aspects of a solution, while not compelling any subdivision of functionality, e.g., into a policy or analytics that are somehow separate.

Transport networks have been evolving with the advent of SDN, and prior to SDN new transport devices support many of Southbound Interfaces (SBI) protocols like Path Computation Element Protocol (PCEP), TL1, SNMP, CLI, XML, NETCONF, OpenFlow, et al. The application of SDN and the use centralized controllers to interface with transport devices using protocols on its SBI, making the southbound fragmented. Domain controllers abstract the fragmented southbound view for its northbound clients by normalizing the NBI across various technologies, protocols, and vendors.

Transport APIs would allow the transport domain controller to communicate with the orchestrator via the normalized NBI to automate and programmed end-to-end transport resources, leveraging the transport infrastructure in an optimized way, across single or multi-domain technologies, and multiple SBIs.

## 2.4 Transport Northbound Interface

Firstly, Northbound interfaces (NBIs) can be organized in two broad categories:

The first category contains low-level information modeling NBIs. The primary role of an Information Model is to converge state representation of data plane devices and abstract the heterogeneity of forwarding technology. Network information models have been developed before the introduction of the SDN paradigm by multiple formal and information SDOs, like the ITU.

Relevant to the SDN paradigm is the ONF information modeling working group (WG), which develops the Common Information Model (Core Model) specifications.

The Core Model is hierarchical and includes a core model, which provides a basic abstraction for data plane forwarding elements, and a technology forwarding and an application-specific model, which evolve the core model abstraction. Core Model specifications exploit object inheritance and allow control applications to acquire abstract network connectivity information and, in parallel, access technology-specific attributes of network elements.

The second NBI category contains high-level and innovative control abstractions, exploring interfaces beyond the typical match-action-forward model of newer forwarding technologies. These interfaces are typically implemented as SDN management applications, use the information model to implement their control logic and are consumed by external entities, like the OSS, the service orchestrator and other control applications.

Effectively, both interface types manifest themselves between the functional interfaces between the Network and Service Orchestrator components.

Any NBI will require resource models these are being developed in formal and informal SDOs, including: IETF, ONF and MEF; which can be used on the interfaces of a domain controller and an orchestrator. Each domain controller and orchestrator can use models developed by different SDOs. Therefore, it is important to ensure that all models support deployment use cases and related functionalities to allow a seamless translation and mediation between systems using different models.

## 2.5 Defining the Transport Northbound Interface

A transport network is a server-layer network designed to provide connectivity services, or more advanced services like Virtual Private Networks (VPN) for a client-layer network to carry the client traffic opaquely across the server-layer network resources. It acts as a pipe provider for upper-layer networks, such as IP network and mobile networks.

Transport networks, such as Synchronous Optical Networking (SONET) and Synchronous Digital Hierarchy (SDH), OTN, WDM, and flexi-grid networks, are often built using equipment from a single vendor and are managed using private interfaces to dedicated EMS and NMS. All transport networks have high benchmarks for reliability and operational simplicity. This suggests a common, technology-independent management and control paradigm that is extended to represent and configure specific technology attributes.

The need for operators to manage multi-vendor and multi-domain transport networks (where each domain is an island of equipment from a single supplier) has been further stressed by the expansion in network size. At the same time, applications such as data center interconnection require larger and more dynamic connectivity matrices. Therefore, transport networks face new challenges going beyond automatic provisioning of tunnel setup enabled by GMPLS (Generalized Multi-Protocol Label Switching) protocols to achieve automatic service provisioning, as well as address opportunities enabled by partitioning the network through the process of resource slicing. With lower operational expenditure (OPEX) and capital expenditure (CAPEX) as the usual objectives, open interfaces to transport networks to meet these requirements. Again, the concept of SDN mentioned earlier leverages these ideas.

The YANG modeling language is the data modeling language of choice within the IETF and has been adopted by several industry-wide open management and control initiatives. YANG may be used to model both configuration and operational states; it is vendor-neutral and supports extensible APIs for control and management of elements.

There are several scenarios where an open interface to access server-layer (transport) network resources would be useful. For the data centre operator, assuming their objective is to trigger the transport network to provide connectivity on demand, the following capabilities would be required:

• Acquisition of the topology, be it physical or logical, of the transport infrastructure resources;

• The ability to obtain information about a set of access points of the transport network facing the client side, including information such as access point identifiers, capabilities, location, and environment types (Data Centers, Storage, et al.);

• The capability to send a request for a service using the access point information, as well as the ability to retrieve a list of service requests and status: source nodes, destination nodes, and current bandwidth and service attributes;

• Telemetry and monitoring of network performance information for real-time monitoring and optimization.

Each of these will require management and control via open interfaces for multi-domain networks with homogeneous technologies (such as OTN), but it can be extended further to multi-domain networks with heterogeneous technologies with higher complexity.

## 2.6 Core Requirements for the Transport Northbound Interface

### 2.6.1 Generic Requirements

**User Intent:** Transport models should maintain separation between high-level user intent and the operational state of the network; e.g., maintain separation between user service request, including all constraints, and the actual service and connection state in the network.

**State Management:** Network and service objects should support the following states: administrative state, operational state, and lifecycle state. Administrative state and operational states are well understood. Lifecycle state is defined by the ONF to model the following entity lifecycle states: planned state, potential state, installed state, in conflict state, and pending removal state.

**Identifiers:** Network and service objects and would include a unique entity ID provided by the controller. The identifier would be chosen such that the same entity in a real network topology will always be identified through the same ID, even if the model is instantiated in separate data stores. Controllers may choose to capture semantics in the identifier, for example to indicate the type of entity.

### 2.6.2 Topology Requirements

The model should support the following topological link and node definitions:

* Link Requirements
	+ Abstract Links
	+ Compound Link which are internally aggregated lower level links
	+ Access Links which connect the router port to the client port of the transport system
	+ Transitional Links which provide adaptation capability between layers within a network element
* Node Requirements
	+ Physical Node
	+ Abstract Node
	+ Chassis / Forwarding Domain

The Link should support various link related attributes like cost, latency, capacity, risk characteristics (including Shared Risk Link Groups - SRLGs). The model should provide clear association between Link and its topology (including virtual topology), nodes and termination points.

* Connectivity Service
* Path Computation
* Notification

The model should be able to provide association between the Link and any underlay circuit / service supporting the Link.

### 2.6.3 Telemetry Requirements

Topology service clients (which in the T-SDN context could be various: applications, orchestrators, controllers, big data collectors, analytics processors, network planners, etc.) require accurate real time network state information (this is known as network telemetry).

Telemetry information will be instrumental for maintaining network efficiency and optimal control under failure conditions. Network telemetry streams would provide resource failure prediction across network resources and provide knowledge to route the provided transport connectivity services away from predicted failure areas; identify and predict points of congestion and eliminate and/or mitigate the congestion by deploying extra network capacity in a timely manner. Clearly network telemetry is a valuable source of information useful for network planning, troubleshooting and resource optimization, and will require suitable models, such as “YANG models for ACTN TE Performance Monitoring Telemetry and Network Autonomics” [9]

# 3. Transport Services

Transport networks are generally designed to deal with “connections” or “services”, which are entities that encompass multiple related optical forwarding technologies.

The transport orchestrator needs to be capable to request service connectivity from the transport controller to support application and/or IP routers connectivity. The type of services could depend of the type of physical links (e.g., OTN link, WSON link, ETH link or SDH link) between the routers and transport network.

# 4. Applicability of YANG to Transport Northbound Interface

The transport NBI data models will required for representation of objects that can be configured or monitored within the transport system. Within the IETF, YANG [10] is the language of choice for documenting data models, and YANG models have been produced to allow configuration or modelling of a variety of network devices, protocol instances, and network services. YANG data models have been classified in [11] and for services in [12].

# 5. Conclusion

A variety of industry challenges remain for the development of transport API orchestration technologies. Emerging protocol and model solutions discussed in this paper are immature and will require further investigation and development before they can be operationalised and used by operators.