Demonstration of Extensible Threshold-Based Streaming Telemetry for Open DWDM Analytics and Verification

Abhinava Sadasivarao[†], Sharfuddin Syed[†], Deepak Panda[§], Paulo Gomes[♛], Rajan Rao[†], Jonathan Buset[†], Loukas Paraschis[†]
Infinera Corporation, [†]Sunnyvale CA, USA, [§]Bangalore, India and [♛]Lisbon, Portugal
Jag Brar*, Kannan Raj[‡]
*Oracle Cloud Infrastructure, *Seattle, WA and [‡]*San Diego CA, *USA*

Abstract: A novel and practical threshold-based extension of streaming telemetry that advances open WDM analytics and introduces network verification, is demonstrated employing an extensible NOS application agent combined with standard NETCONF/YANG and open-source software technologies. © The Author(s).

1. Motivation and OFC Relevance

We demonstrate a novel and practical *threshold-based* extension to streaming telemetry that advances open WDM analytics. Specifically, extensible network operating system (NOS) agents are developed for network element (NE) and network-wide performance monitoring (PM) applications, employing well-established NETCONF/YANG and open source software technologies. In addition to PM and analytics, these agents also enable the introduction of, for the first time to our knowledge in optical transport, network verification, which determines if the network can continue to operate within desired correctness criteria. We validate the proposed non-commercial research prototypes and associated innovations through live demonstration of three different use-cases on a multivendor WDM testbed.

The main motivation for this work, and its relevance to OFC and the broader optical networking community, arises from the strong interest of next-generation transport networks (especially inter-DCI transport) in model-driven networking (MDN) [1,2] streaming telemetry [3], analytics [4, 5], and more recently also declarative configuration management (DCM) [6]. Streaming telemetry is fast becoming the *de-facto* approach to PM for both packet and optical transport networks, replacing the legacy polling-based SNMP frameworks. Our design and demonstration add significant enhancements to the existing streaming telemetry frameworks [1-3, 5] by introducing threshold-based PM which can *adaptively adjust* the frequency of streaming and/or the thresholds. Moreover, this work introduces network verification, which has become increasingly important in networking but to our knowledge has not before been demonstrated in optical transport. Network verification detects potential misconfigurations [7] before they are applied on the NE, thus preventing degradation of network performance, availability, or security. In this sense, network verification is particularly important for DCM as such an *a priori* verification can prevent immediate or in some cases eventual errors (e.g. addition of a new BGP neighbor causing routing loops). The importance of network verification has increased significantly in recent years due to the emergence of "SDN" automation and programmability which has substantially expanded the scope for misconfigurations, either human or machine driven. Introducing network verification to optical transport, we believe, is particularly applicable as the latest generation DWDM systems adopt open transport, programmability and disaggregation, while also scale ever closer to the fiber Shannon limit by optimizing the reach-capacity tradeoff with increased granularity [8].

2. Demonstration Innovations in Streaming Telemetry Design and Applications

A typical streaming telemetry implementation employs *push-based* subscriptions to a collector that monitors at specified *sample intervals* NE metrics such as transponder Rx/Tx optical power, Q, BER, CD, optical amplifier pump-power, or CPU utilization. Currently, OpenConfig and GNMI have become the most popular choice of vendor-neutral telemetry framework leveraging YANG based MDN [1-3]. However, two potential challenges have recently been associated with such telemetry frameworks: (a) The monitored values within a subscription are all *normalized* to be streamed at the same, constant frequency (b) The state of the system is not taken into consideration. Hence, threshold-based telemetry has been considered a potential advancement in network PM. To this end, we define and demonstrate a telemetry mechanism that adds two new parameters to control the rate of streamed data: (i) Configurable thresholds $Th_{i=1..n}$ for each of the 'n' monitored metrics, and (ii) One or more policies $P_{j=1..m}$ for each metric that updates the streaming frequency when a specific Th_i value is met. This extended framework allows higher resolution PM in face of degradation, potentially before impending faults. For example: while a WDM transponder might be configured in steady state to push telemetry updates on optical Rx power (OPR)

once every 5 seconds, if the OPR starts degrading, it may be useful to monitor the OPR variations more closely by

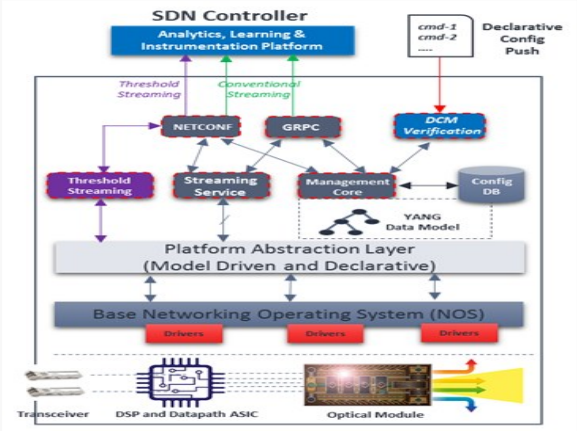

Fig. 1 Overview of NOS Software Architecture

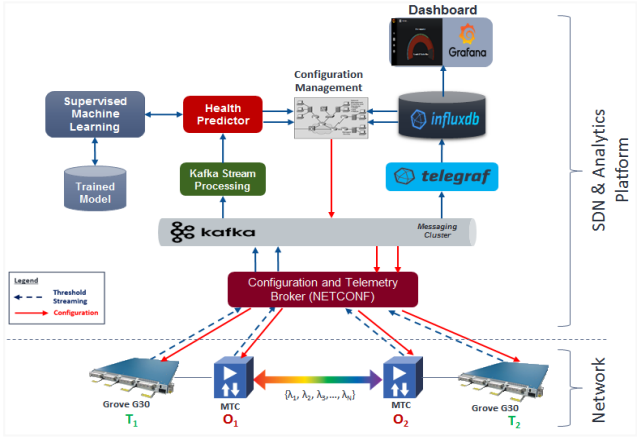

Fig. 2 Demonstration Testbed Setup

streaming faster. Our threshold-based streaming framework can define a policy that would increase the streaming frequency when power gets out of a certain range. Additional policies can exist for the same subscription to allow PM correlations for further analysis; e.g. in addition to OPR also stream Pre-FEC BER. Th_i and P_j can be configured by the network operator or rely on the defaults of the system vendor. Note that our design and demonstration is intentionally agnostic and extensible to setting and updating Th_i and P_j, because we consider this process to be a separate very interesting and novel area of research, especially when such an advanced streaming methodology may leverage adaptive thresholds, potentially combined with machine-learning (ML) techniques towards predictive analytics [2-5, 8, 10]. Finally, note also that threshold streaming is better than *legacy threshold crossing alerts* (TCA), which are standing conditions raised when a specific threshold is crossed (and cleared when the monitored value falls below the threshold), because while the condition is active, legacy TCAs provide no further indications on the rate of variation of the PM metric.

Furthermore, our telemetry enhancements enable also network verification, which can be particularly useful when DCM is desired by the operator as the means to realize intent-based configuration. DCM significantly simplifies network state management allowing operators to only track global, network-wide intents/policies. In this sense, DCM is typically built upon *full configuration push* which allows an "SDN" Controller to operate on global policies without maintaining the individual, per-device configuration (and operational) state. Therefore, leveraging a more granular monitoring enabled by our current threshold-based streaming, the SDN Controller can more accurately track the current state of the network, which in turn reflects the *runtime profile* of the NEs and thus the NE's ability to apply/accept a declarative configuration push. This can be factored into the Controller's device configuration generation, as an important *pre-validation* step of DCM. Note that a failed or an incomplete declarative config could result in several hours of downtime, especially as current cloud-scale operators maintain million lines of device configuration commands, and often deal with thousands of configuration changes per day [6]. Note also that DCM system verification is applicable to both data and control plane system aspects; e.g. A recent FCC report [6] which analyzed a major North American network outage, prescribes specifically system memory/CPU utilization PM alarms to improve early detection and calibration. Using threshold streaming and network verification, in this case, the management systems (OSS/BSS) could have observed early degrading trends and avoided any subsequent device configuration updates that would compromise the network stability. Thankfully, new open-source network verification tools are becoming increasingly available [9].

3. Demonstration Overview

We demonstrate optical network threshold-based streaming and verification, building upon the recent work of an extensible NOS based on an MDN & DCM framework for open DWDM systems [3, 5]. The software in this demonstration however extends beyond the embedded optical NOS to components in the Controller/Analytics layers. To validate our research prototypes, our demonstration (Fig. 2) implements three different use cases in a live multivendor WDM testbed employing multiple DCI transponder and line system NEs, which use at least three different embedded NOS types. Conventional streaming over GRPC is natively implemented as part of the management plane of the embedded NOSs. To support threshold streaming, an additional native subsystem (Fig. 1) directly interfaces with the underlying platform abstraction layers to monitor the state of each NE at a finer

resolution and to ensure accuracy when determining the threshold crossing triggers. The threshold-based subscriptions are created using NETCONF's standard **`<create-subscription>`** RPC. Updates are streamed using NETCONF **`notifications`**. We utilize specifically **openconfig-system.yang** to monitor the NE's control-plane resources and stability (NOS per-process statistics on CPU, system load, memory and I/O utilization), and **openconfig-platform.yang** and **openconfig-terminal-device.yang** to monitor the transponder's optical channel "data-plane" attributes (CD, PMD, Rx/Tx power, Q, BER, client-side Ethernet PMs etc.). A common NETCONF server on the NE (Fig. 1) supports both configuration and streaming, as NETCONF offers rich streaming protocol features (e.g. SSH for security, reliable connection-oriented sessions over TCP), and robust configuration functionality making threshold updates most convenient for operators [2]. Any bandwidth penalties from XML serialization can be overcome by encapsulating the streamed PM data in other wire-efficient encoding (such as ProtoBuf or Thrift) within the XML body using NETCONF's **`binary`** (base64) construct.

To demonstrate network verification, the NE provides a *DCM verification (DCMv)* service, based on a NOS agent running on the NE, that *pre-validates* prior to accepting configurations updates. This agent can ingest full *declarative configuration push*, specified either as a set of CLI commands, or as a NETCONF XML payload, and can parse and validate the declarative configuration received by the NE (via *update*, *create* or *delete* operations). The delta configuration changes to be applied are identified using a *mark-and-sweep* approach which walks through the NE's object database. The DCMv agent compares the current NE state against the (known) steady state before an NE accepts (or rejects) the configuration change. Through a *closed-loop system,* the SDN Controller can act upon the observed data and determine the possible actions to be triggered on the NEs. The controller analytics (Fig. 2) employs open-source software, specifically an *Apache Kafka* cluster is ingested via *Telegraf* stream *consumer* pipeline to archive the streamed PM metrics into *InfluxDB* time series database which is visualized in real-time using *Grafana* dashboards. We finally integrate *Tensorflow* (an open-source ML framework) to obtain a univariate regression model of the NE CPU utilization, and then estimate a *health score (HS)* for the NEs. The *HS* value ranges from 1 to 100 and varies inversely proportional to the CPU utilization. Towards minimizing potential NE instability, the configuration manager monitors the *HS* and can take corrective steps (e.g. decrease scheduling priority of inessential NOS processes) in order to maintain network stability. Our live demonstration testbed would showcase a few important examples focusing particularly on the following use-cases:

1) *Optical ("data-plane")* monitoring of the per-carrier Q, BER and Tx/Rx power, which each NE streams initially ("factory-default") every 30s but during degradation accelerates in 2.5s steps to an upper limit of 1s.
2) "*Control-plane"* monitoring of each NE CPU load averages, which each NE streams on a per-NOS process granularity, providing visibility on impact of the different active software. Again, streaming initially occurs every 30s but once CPU load exceeds a pre-defined threshold streaming gradually increases in steps of 2.5s.
3) Network verification resulting to the DCMv agent rejecting the configuration changes a) during a declarative configuration push to a transponder operating in a degraded OPR state, and b) during new streaming subscription requests to an NE operating at higher than desired CPU load.

To induce high NE CPU utilization, we use **stress-ng** which cycles through several synthetic computations. To simulate WDM degradation, a VOA is inserted between the transponders and the line-system.

In conclusion, we demonstrate a novel and practical *threshold-based* extension to streaming telemetry that advances open WDM analytics and introduces network verification in optical transport, employing an extensible NOS application agent combined with standard NETCONF/YANG and open-source software technologies. Our current framework can also extend to "adaptive" thresholding, leveraging also ML techniques, towards the increasingly interesting potential for predictive analytics for network planning, control, and operations [8, 10].

4. References

[1] A. Shaikh et al., "*Vendor-neutral Network Representations for Transport SDN*", in OFC (2016), A. Shaikh, "*Multi-vendor Streaming Telemetry*", in OFC, (2018), M. Birk, "*Open Platforms for Optical Innovation*", in OFC, (2018).
[2] V. Dangui, "*Key Enablers of Automated Optical Networks*", in OFC (2018) and M. Machacek, "*Network Monitoring for Cloud*", in OFC (2018) and M. Rizzi, "*Automation of Optical Provisioning on Multi-Vendor Metro Optical Platforms*", in OFC (2017).
[3] F. Paolucci et al. "*Network Telemetry Streaming Services in SDN-Based Disaggregated Optical Networks*", JLT (2018) and A. Sadasivarao et al., "*High Performance Streaming Telemetry in Optical Transport Network*", in OFC (2018).
[4] R. Singh et al., "*RADWAN: Rate Adaptive Wide Area Network*", in ACM SIGCOMM (2018).
[5] A. Sadasivarao et al., "*Demonstration of Advanced Open WDM Operations and Analytics, based on an Application…*", in OFC (2019).
[6] B. Koley, "*The Zero Touch Network*", in IEEE CNSM (2016) and CenturyLink Network Outage Report, FCC (December 2018) https://docs.fcc.gov/public/attachments/DOC-359134A1.pdf.
[7] A Fogel et al., "A General Approach to Network Configuration Analysis", in USENIX NSDI (2015).
[8] L. Paraschis et al., "*System Innovations in Inter Data-Center WDM Transport Networks*", in ONDM (2019).
[9] ONF Next-Generation SDN (NG-SDN), https://www.opennetworking.org/reference-designs/ng-sdn/, Veriflow, https://www.veriflow.net/ and Batfish (https://www.batfish.org/).
[10] L. Paraschis et al., "*Proactive …*" US Patent 10038494 and "*Multi-Layer mechanisms…*" US Patent Application 15/917,386.