

# CASTOR: A Monitoring and Data Analytics Framework to Help Operators Understand What is Going on in Their Networks

A. P. Vela <sup>(1)</sup>, Ll. Gifre <sup>(2)</sup>, O. González de Dios <sup>(3)</sup>, M. Ruiz <sup>(1)</sup>, and L. Velasco <sup>(1)\*</sup>

<sup>(1)</sup> Universitat Politècnica de Catalunya, Barcelona, Spain, \*lvelasco@ac.upc.edu

<sup>(2)</sup> Universidad Autónoma de Madrid, Madrid, Spain. <sup>(3)</sup> Telefonica, Madrid, Spain

**Abstract** *CASTOR MDA allows collecting huge amounts of measurements from devices. Such data needs to be preprocessed and analyzed to discover and summarize knowledge that can be consumed by human operators. Helper tools are demonstrated on a realistic multilayer network scenario.*

## Introduction

The increasing network complexity together with that of traffic dynamicity and quality and availability guarantees, affects costs and reduces margin to network operators. From the operational viewpoint, several improvements need to be implemented to greatly reduce or even virtually eliminate the impact on the services that results from defects and gradual degeneration and aging of network devices. These improvements need to come from anticipating such degradations instead of waiting for alarms being triggered after some thresholds are exceeded. Anticipation requires monitoring network connections and devices, and analyzing collected measurements, while correlating them with operational databases, i.e., topology and connections, and ideally, with services databases, e.g., for SLA verification.

In this demonstration, CASTOR distributed Monitoring and Data Analytics (MDA) architecture [1] is shown. CASTOR consists of: *i) MDA agents running closed to the network devices*, which collect monitoring data records, pre-process them, and apply data analytics algorithms to discover knowledge from data (KDD); and *ii) a centralized MDA controller* that applies network-wide data analytics algorithms, while correlates data with operational and service databases. Discovered knowledge can be applied for several purposes, including proactive network reconfiguration thanks to the interaction with state-of-the-art software-defined networking (SDN) controllers, e.g., ONOS, and planning tools, e.g., Net2Plan [2]. In addition, data analytics algorithms can be devised to help human operators.

In this demonstration, three different tools to assist network operators will be shown: *i) network traffic evolution and projection* as a specific tool for network planning purposes; *ii) soft-failure localization* affecting lightpaths [3]. An interface with ONOS will be shown for re-routing affected lightpaths excluding the failed resource, where routes will be computed in Net2Plan; and *iii) a visualization process* that analyzes bit error rate (BER) monitoring data collected for all the lightpaths in the network

(>800) in the last month (one measurement every 15 minutes). The interactive visualization process presents data in a way that human operators can easily identify the cause of a degradation affecting a subset of lightpaths [4].

## CASTOR architecture

The CASTOR MDA architecture shown in Fig. 1 consists of a multi-node agent, named MDA agent, managed by the centralized MDA controller in the control and management plane. The number of MDA agents varies depending on the size of the network, geographical extension, and/or any other criteria.

MDA agents are designed to configure monitoring and telemetry, as well as to collect measurements from one or more nodes in the data plane. While each node controller usually controls one single node and exposes one single interface toward the SDN controller, MDA agents are designed to be in charge of monitoring and telemetry of a set of nodes. MDA agents consist of two building blocks, the local configuration module and the local KDD module. The local configuration module is in charge of receiving configuration and exposes a RESTCONF-based North-Bound Interface (NBI), based on a YANG data model, to the MDA controller. Finally, a number of node adapters (one per physical node) are used to implement the specific protocols exposed by every node controller for node configuration and for monitoring data collection.

Although we assume that node controllers' NBIs are based on a common YANG data model, different protocols for configuration, monitoring, and telemetry might be considered (e.g., NETCONF, IPFIX, and gRPC-based protocols for configuration, monitoring and telemetry, respectively, NETCONF for everything, or any other combination). For this very reason, node agents include bespoke node adapters implementing specific protocols and function mapping for the underlying node controller.

The local KDD module is intended to apply data analytics to the measurements received from the nodes focused on the KDD process. Output of the data analytics procedure is forwarded to the MDA controller to implement network-wide

control loops; two types of messages are supported: *i*) IPFIX-based monitoring messages including processed monitoring samples (i.e., values are averaged over the selected monitoring period, e.g., 15 minutes); and *ii*) through asynchronous notifications using the RESTCONF NBI. Regarding telemetry, measurements are locally processed by specific KDD processes in the MDA agent to reduce data exchange with the MDA controller. Note that telemetry measurements might be taken in a sub-second basis, so analysis is performed locally in the MDA agent and results can be conveyed to the controller for decision making. The network control and management plane includes the SDN controller and the centralized MDA controller. Data records and notifications from the MDA agents are received in the MDA controller, where a *data manager* module is in charge of storing them into a scalable multi-master replication database, e.g., Apache Cassandra, as well as to notify the *process manager* module, which in turn executes the corresponding KDD process, optionally, on a large-scale data processing engine like Apache Spark. Additionally, the MDA controller manages monitoring and telemetry configuration across network nodes and monitoring devices, as well as into KDD applications deployed in the MDA agents, through a dedicated RESTCONF interface. In addition, operational databases (e.g., topology / traffic engineering -TE- and connections -LSP- databases) synchronization from the SDN controller, as well as topology and LSP update notifications, is supported. With operational databases synchronized, sophisticated procedures can be developed as the CASTOR MDA controller is not only able to

collate measurements from the agents, but also to correlate them with the route of a LSP for failure localization purposes. In case that a network reconfiguration is needed, the SDN controller can be also triggered from the MDA controller.

### Demonstration

This demonstration proposes a multilayer network scenario controlled by a single ONOS instance and CASTOR MDA (Fig. 2). Optical nodes and L2 switches are configured inside ONOS as Null Provider to create an emulated network topology. Several generators replay traces with measurements recorded from real network nodes and exports them to MDA agents for further processing by means of IPFIX protocol emulating network devices with monitoring capabilities. Specifically, we assume that optical transponders are able to measure BER and received optical power, whilst L2 switches measure packet traffic, e.g., received and transmitted bytes and number of Ethernet frames. Besides, optical spectrum analyzers (OSA), installed in egress ports of optical nodes, are able to monitor the spectrum of optical links. The workflow of the demonstration will be as follows: *i*) the MDA controller imports operational databases, i.e., TED and LSP-DB, from ONOS. One new lightpath is created in ONOS and automatically synchronized to CASTOR MDA; *ii*) analysis of historic traffic and future traffic projection is requested for a MPLS service comprising a number of LSPs. The tool extracts daily minimum, mean, and maximum traffic belonging to the selected service and projects those metrics for the next year; *iii*) an operator activates monitoring of a lightpath (transponder

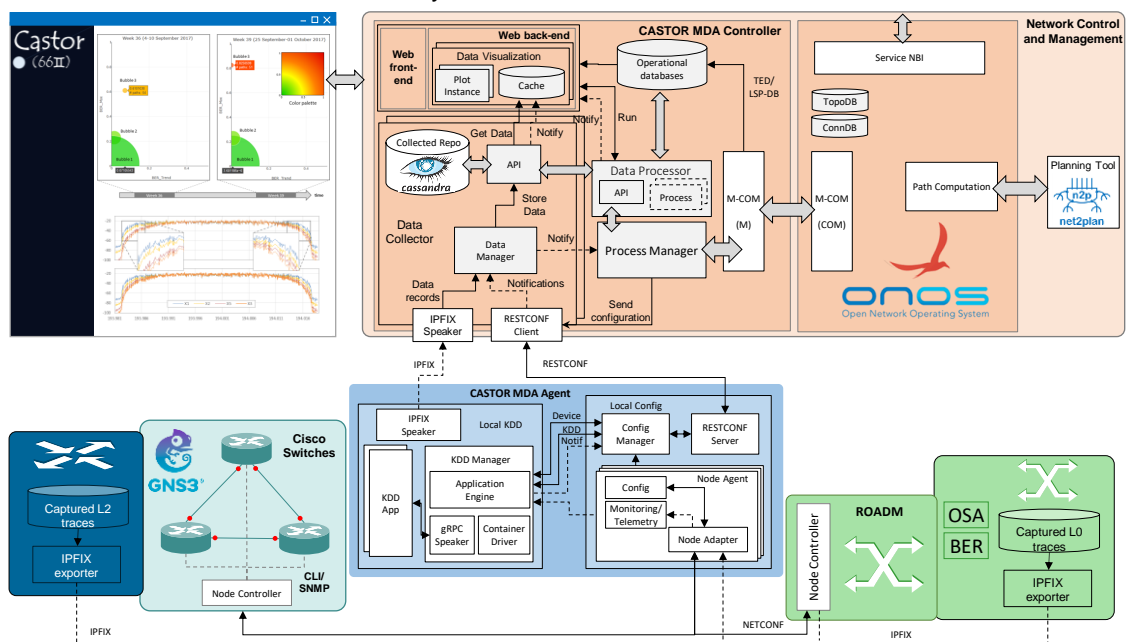


Fig. 1: Demonstration's set-up includes CASTOR's MDA controller and agents together with ONOS and Net2Plan planning tool

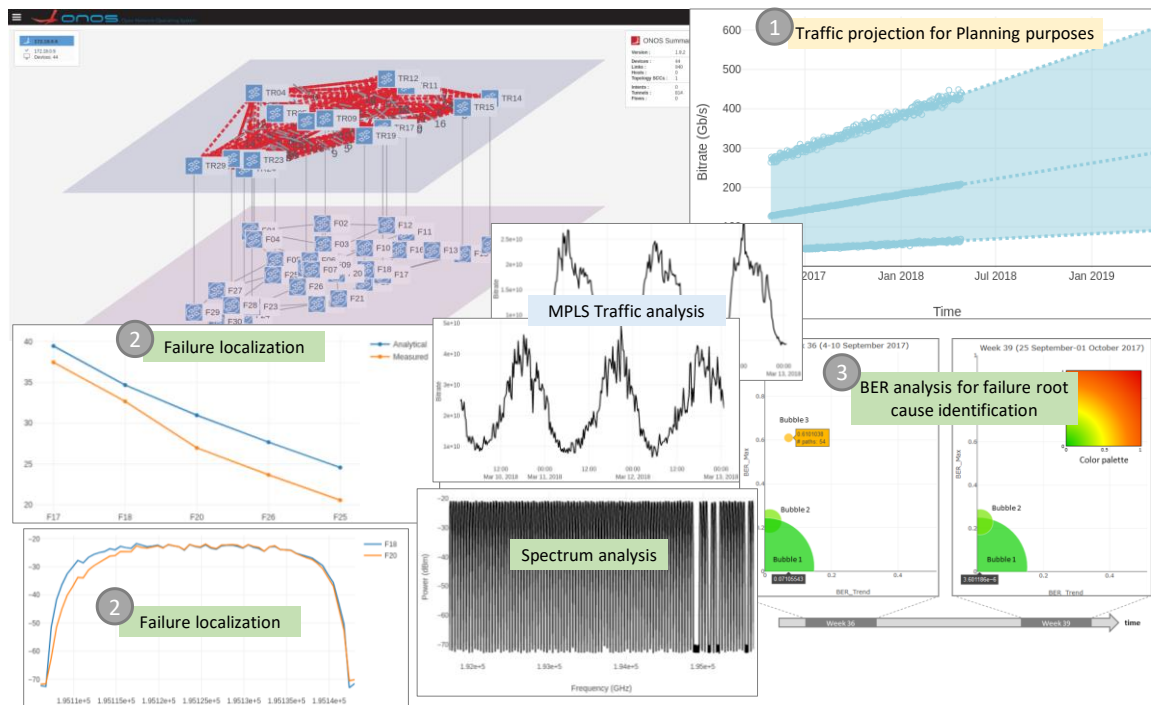


Fig. 2: Multilayer demonstration scenario and three tools and auxiliary graphs to assist network operators

and OSA), an MPLS LSP, and a L2 virtual link. Measurements are received in CASTOR's MDA agents and collated in the MDA controller, where they will be plotted. iv) lightpaths' BER is continuously measured and locally analyzed in the MDA agent; unexpected slight BER variations are detected and notifications are sent to the MDA controller. An operator can initiate a failure localization procedure, where the tool running in the MDA controller interrogates every MDA agent in the route of the lightpath asking to verify the optical signal, as well as to measure the OSNR, both using the local OSA. The centralized tool collates all the measurements to find the failure. Once localized, the operator can trigger the rerouting of the affected lightpath to ONOS excluding the failed resource; v) upon receiving several notifications from the MDA agents regarding a set of lightpaths experiencing unexpected BER, the operator runs a specific data visualization tool that presents BER measurements of the last month using a *bubbles chart* to clearly separate normal lightpaths from those showing unexpected BER behavior in terms of its value and its evolution with time. The operator can select one of the bubbles with the lightpaths of interest and use a *spectrum color map* diagram that suggests several links as the main cause of the failure. Individual inspection of each of these links reveals the root cause of the failure.

### Innovation and Relevance

CASTOR MDA aims at bringing cognitive capabilities to state-of-the-art SDN controllers. In this demonstration, integration with ONOS will be shown. To this end, custom extensions have

been developed for both frameworks extending their base services and interfaces, namely: i) ONOS east-west interface to export TED and LSP-DB, as well as for the MDA controller to send recommendations to ONOS. ONOS is connected through another east-west interface to an instance of Net2Plan for path computation. In addition, CASTOR MDA analyzes huge amount of monitoring data received from the network devices for KDD purposes. In the demonstration, three different applications especially designed for assisting network operators to understand what is going on in their networks will be exhibited.

The demonstration presents timely innovations adding monitoring and data analytics capabilities to existing SDN controllers for operating complex network scenarios. Hence, the demonstration will attract the community's attention and it will be of interest to a broad ECOC audience.

### Acknowledgements

This research has received funding from the EC through the METRO-HAUL project (G.A. n° 761727), the AEI/FEDER TWINS project (TEC2017-90097-R), and ICREA Institution.

### References

- [1] L. Gifre *et al.*, "Autonomic Disaggregated Multilayer Networking," IEEE/OSA JOCN, 2018.
- [2] P. Pavon-Marino *et al.*, "Net2Plan: An Open Source Network Planning Tool for Bridging the Gap between Academia and Industry," IEEE Network, 2015.
- [3] A. P. Vela *et al.*, "Soft Failure Localization during Commissioning Testing and Lightpath Operation [Invited]," IEEE JOCN, vol. 10, pp. A27-A36, 2018.
- [4] A. P. Vela *et al.*, "Applying Data Visualization for Failure Localization," in Proc. OFC, 2018.