The Software Defined Networking in KM3NeT

Tommaso Chiarusi[1,*] and *Emidio* Giorgio[2,**] on behalf of the KM3NeT Collaboration

[1]INFN - Sezione di Bologna, Viale Berti-Pichat 6/2 40127 Bologna, Italy
[2]INFN - Laboratori Nazionali del Sud, Via S. Sofia 62, 95123 Catania, Italy

Abstract. The networking infrastructure of the KM3NeT detector, implemented with both White Rabbit and standard high preformance switches, is presented in its peculiar asymmetric and hybrid layout. It is one of the first use-cases for Software Defined Networking in Astroparticle Physics experiments. Thanks to this innovative technology, dangerous network loops are prevented and a complete deterministic control of the handled data flows is obtained.

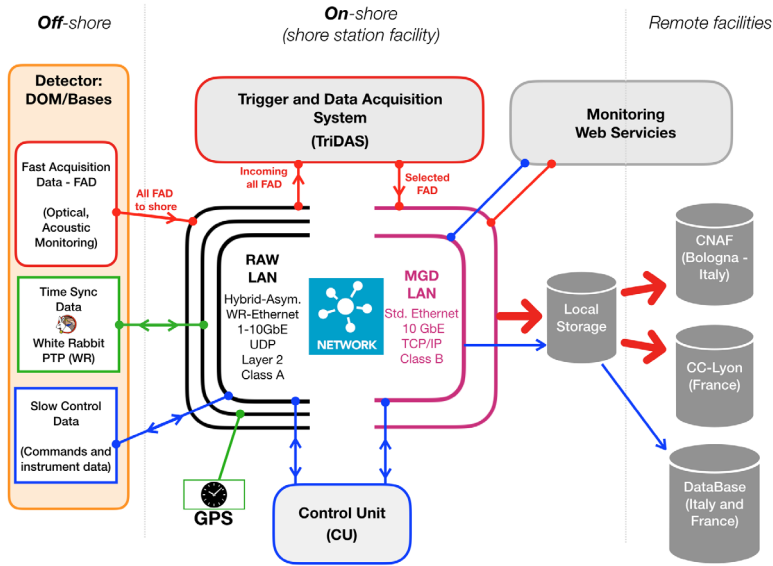## 1 The KM3NeT networking layout

KM3NeT is a distributed neutrino observatory in abyssal sites of the Mediterranean Sea [1]. Each installation will be based on a grid of thousands Digital Optical Modules (DOMs), interconnected to shore via an electro-optical seafloor infrastructure extended over distances of O(100) km. The DOMs are organised in vertical structures, the Detection Units, each one provided with a Base-module for controlling the power supply and optical amplification of the attached devices. Exploiting a custom FPGA-based White Rabbit [2] kernel with Ethernet connectivity, the DOMs and Base-modules are submarine nodes of the global Layer 2 optical networking infrastructure[3]. The network segments for the KM3NeT data acquisition (DAQ) system are represented in the scheme sketched in Figure 1, together with the relevant DAQ elements [4–6]. We focus here on the *RAW LAN*, which is the Layer 2 segment that interconnects the off-shore detector to the Control Unit (CU) and Trigger and Data Acquisition System (TriDAS) servers, on-shore. As anticipated, it has to comply with the two following important constraints:

- **Asymmetry of the connections**: the asymmetry directly originates after the so-called *optical broadcast* architecture adopted for the optical infrstructure. It aims at best exploiting the number of optical fibers contained in the many km-long electro-optical cable (EOC) which connects the shore stations with the detectors. All the timing and slow-control information addressed downstream from the shore station is embedded into a single stream of data. This is split at any next branching step of the downstream connections up to every endpoint off-shore. Despite all DOMs and Base-modules receive the same information but only the actual destination device process it.

- **Hybrid switching layout**: the asymmetry described above violates the requirements of the WR protocol. A customization of the White Rabbit Switch (WRS) is then necessary to optimise and adapt the usual master-slave relation between the WRSs and their endpoints. The

---

*e-mail: tommaso.chiarusi@bo.infn.it
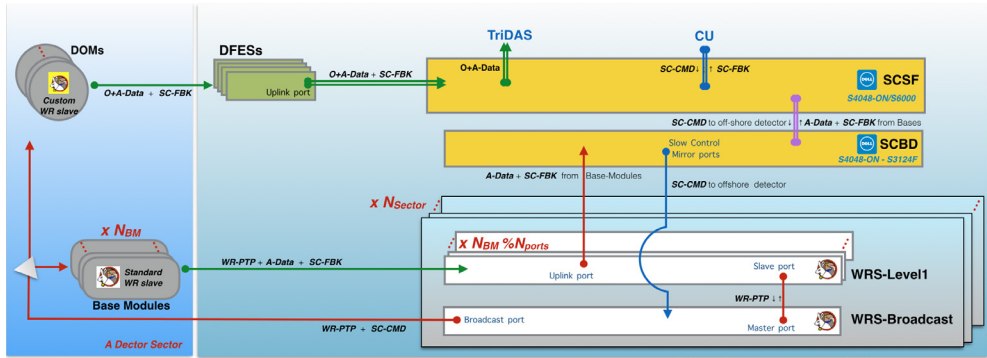**e-mail: emidio.giorgio@lns.infn.it

Figure 1. Sketch of the two KM3NeT network segments for the raw and managed data flows *RAW* and *MGD* LANs, respectively, with the basic elements off-shore, on-shore and in the remote locations. For each network segment, the displayed layers are intended to help the reader to differentiate the different data flows.

WRS-Broadcast and the WRS-Level1 share the mastership versus the off-shore endpoints. This architecture was developed by the Seven Solutions Company [7]. Base-modules are kept in a fully closed WRS loop, while the DOMs do not reply with any WR-PTP packets, and send the Fast Acquisition data directly on standard switches.

The switching infrastructure is composed of the following elements (see Figure 2 for reference):

- **the White Rabbit switch fabric**: a number of WRS-Broadcast and WRS-Level1, dimensioned to cope with the subdivision of the off-shore detectors in convenient different sectors;

- **the Standard Switch fabric**: a number of *DOM Front End Switch* (DFES) elements, dimensioned to host the total number of DOMs; one *Star Center Switch Fabric* (SCSF) element, which interconnects the TriDAS resources and forward to them the optical and acoustic data flows; one *Slow Control and Base Data* (SCBD) element, which is responsible to connect the standard switch fabric with the White Rabbit one.

The sketched connections topology is the simplest and most effective one. Experience demonstrated that to preserve the stability of the WR infrastructure, and consequently grant a reliable time-synchronisation mechanism, it is mandatory to minimise the traffic passing through the WRS-Broadcast. This is achieved with the Software Defined Networking (SDN), a frontier technology applicable by modern high-level switches. With SDN, it is possible to define the most convenient relation between a particular data flow and the due switch ports, simply via the Layer 2 routing. A group of ports of a switch driven via SDN is called SDN instance.

Figure 2. The complete KM3NeT hybrid switching infrastructure in the exploited Software Defined Networking (SDN) layout. Single or double lines represent 1 and 10 GbE connections, respectively. Single directional data-flows are indicated with arrows; bi-directional connections have only dots. The data flow notation is: *WR-PTP*: the WR implementation of the Precision Time Protocol (PTP); *SC-CMD, SC-FBK*: the slow control commands and feedbacks exchanged between the CU and the detector; *O + A-Data*: the optical and acoustic data, respectively. Note that the Base-modules don't produce any optical data. The WRS fabric is dimensioned to match the number of sectors N_{Sector} into which the off-shore detector is conveniently subdivided. The SDN infrastructure mirrors the downstream flow originating from the CU to each WRS-Broadcast.

2 The Software Defined Networking implementation

The core of the SDN implementation is the OpenFlow protocol. It gives access to the forwarding plane of a network switch. OpenFlow allows the user to define the needed forwarding/routing rules, called also flows, and implement them into the SDN switch. In KM3NeT, the SDN-flows are all Layer-2 based, i.e. they rely only on the source and destination MAC addresses written in the packets and/or to the source/destination switch port. If a packet does not fulfill the rules, it is dropped. OpenFlow v1.3 was chosen in order to exploit the MAC-masking feature for handling multiple MAC addresses within one single rule. OpenFlow v1.3 has also the advantage that the injected flows are made persistent in the memory of the SDN switch. Not all the switches can implement the OpenFlow protocol. In KM3NeT the DELL Series S switches (S6000, S4048-ON and S3124F) were selected for they allow to fully exploit the OpenFlow features and keep the possibility to configure the speed and the auto-negotiation of the in-band interfaces. This is important for correctly interfacing the DELL switches to the WR devices.

To handle all the data-flows implied with the KM3NeT Data Acquisition, only a restricted number of SDN rules are required for both SCSF and SCBD instance (see Table 1). It is important to note that the number of rules does not depend on the size of the detector, i.e. on the number of DOMs and Base-modules, thanks to the fact that the MAC addresses of the off-shore nodes always have the *08:00:30* prefix. Finally, failover procedures can require additional rules. The most effective one implements some drop actions, in order to temporarily remove unwanted data-flows.

All the rules are handled by a *Controller* service, which is an additional new element in the network. Any SDN instance continuously polls the Controller (over a TCP connection) in order to update the set of active rules. The rules management is done by the Controller via

Rule #	Source MAC	Destination MAC	Action
SCSF-1	any	ff:ff:ff:ff:ff:ff	to DHCP server
SCSF-2	08:00:30:00:00:00/ff:ff:ff:00:00:00	Control Unit	SC-FBK to CU
SCSF-3	08:00:30:00:00:00/ff:ff:ff:00:00:00	TriDAS Front End	O+A Data to TriDAS
SCSF-4	CU	08:00:30:00:00:00/ff:ff:ff:00:00:00	SC-CMD to SCBD
SCBD-1	08:00:30:00:00:00/ff:ff:ff:00:00:00	any	SC-FBK + A-Data to SCSF
SCBD-2	Anything from SCSF uplink	08:00:30:00:00:00/ff:ff:ff:00:00:00	SC-CMD to multiple WRS-Broadcast
SCBD-3	08:00:30:00:00:00/ff:ff:ff:00:00:00	ff:ff:ff:ff:ff:ff	to SCSF

Table 1. The SDN rules for both SCSF and SCBD instances. The MAC address prefix *08:00:30* is always the same for DOMs and Base Modules, which differentiate for the next 3 bytes digits. The *Action* column summarises the flow handling as sketched in Figure 2.

a dedicated software called OpenDaylight (ODL) [8] which is an OpenFlow implementation with a consistent support from a wide range of hardware vendors. The version currently used in KM3NeT is Nitrogen (0.7.3). ODL allows SDN flows management through a REST-API interface. This interface has been exploited in KM3NeT not only for a high-level management of rules, but also to implement auxiliary customized services, that access the Controller via RESTCONF and produces statistics related to the network packets matching the various rules. This is relevant for monitoring the SDN system and possibly intervene, even automatically, with failover strategies.

3 Conclusions and outlooks

The Software Defined Networking technology was succesfully introduced in the KM3NeT DAQ system. If not unique, it represents one of the few real use-cases of SDN implementation for experiments in the High Energy Physics field. The use of SDN highly improved the data taking stability and fixed important issues of the network, due to matching standard switches with White Rabbit devices.

Beside the extremely important positive results, SDN is still a frontier technology: open-source applications implementing user-friendly configuration and maintenance tools are globally missing. The current goal of the KM3NeT Collaboration is to finalise a custom SDN framework, compliant with the OpenDaylight software, for better controlling and maintaining the networking infrastructure of the neutrino telescopse for decades of lifetime.

References

[1] S. Adrián-Martínez et al., J. of Physics G: Nuclear and Particle Physics **43**, 084001 (2016)
[2] *The open hardware white rabbit group*, http://www.ohwr.org
[3] D. Real, D. Calvo, Physics of Particles and Nuclei **47**, 918 (2016)
[4] C. Pellegrino, T. Chiarusi, EPJ Web of Conferences **116**, 05005 (2016)
[5] R. Bruijn, *KM3NET: Readout and Triggering*, in *these proceedings* (2018)
[6] C. Bozza, *KM3NET: Acquisition Control*, in *these proceedings* (2018)
[7] *The seven solution company*, http://sevensols.com
[8] *The opendaylight project*, https://www.opendaylight.org