Efficient Network Monitoring for Large Data Acquisition Systems

Authors: Dan Octavian Savu (1)
Ali Al-Shabibi (1,2)
Brian Martin (1)
Rune Sjoen (3)
Silvia Maria Batraneanu (4)
Stefan Stancu (4)

(1) CERN, Geneva, Switzerland
(2) University of Heidelberg, Germany
(3) University of Oslo, Norway
(4) University of California, Irvine, USA

Introduction

At the core of the ATLAS DAQ infrastructure there are 3 distinct computer networks responsible for the data transfers between the system's subcomponents. More than 200 switches and routers interconnect around 3500 hosts to build a real-time filtering system for particle collision events.

For the ATLAS Networking Team the operational goals are to prevent network downtime and to be able to track down ad-hoc or post-mortem network issues as fast as possible. A complex software solution has been developed to help networking experts accomplish their goal while providing relevant and up-to-the-minute system information for other related DAQ teams.

The network monitoring software is designed as a modular solution around a central database (N-CDB). The N-CDB acts as a shared data structure for the various modules and is kept up to date by a set of modules performing topology discovery and statistics collection.

The information is accessible either from a web-based user interface or through a custom ATLAS Data API Mechanism (ADAM) for data exchange. Additionally, a self-check mechanism warns about any module not functioning as expected, by sending detailed mail messages to experts.

Topology Consistency

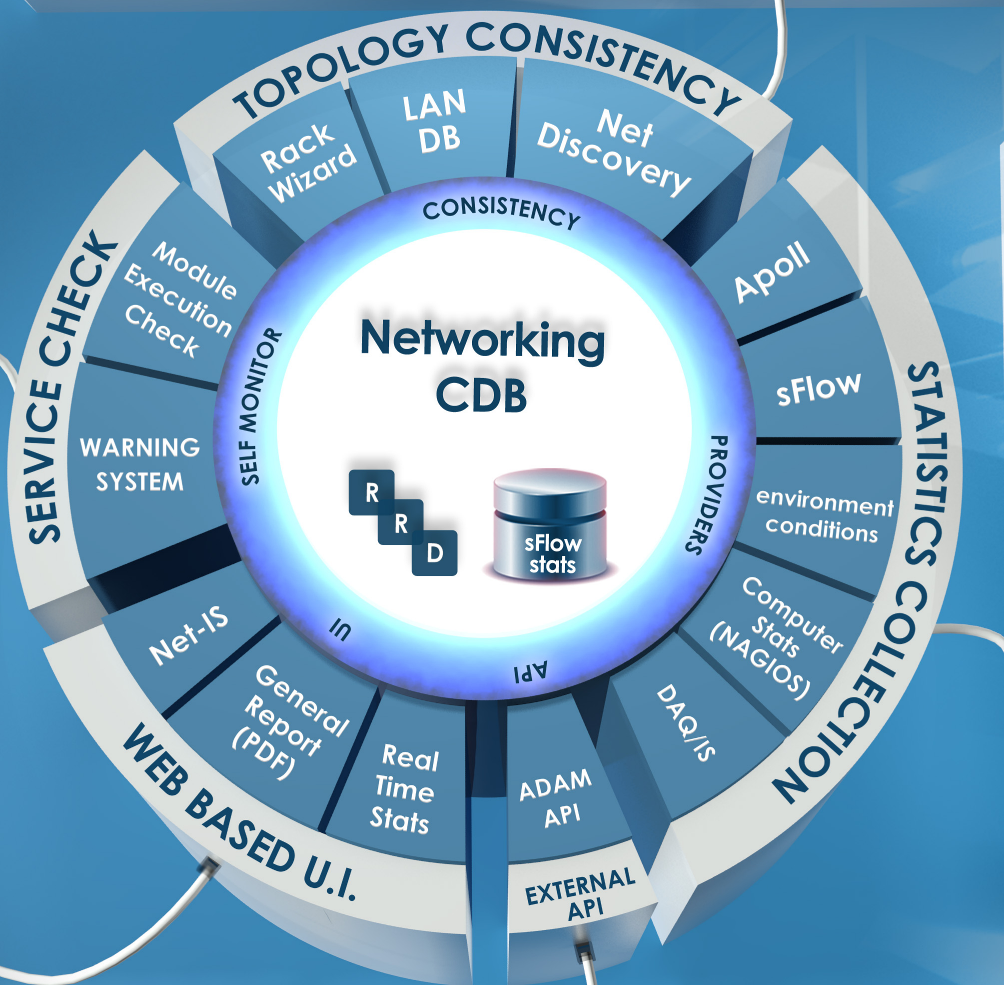
An accurate representation of installed devices and network connections in the N-CBD is assumed by every module and is critical for a normal behaviour of the system. To keep the topology representation up to date, a full network discovery process is run periodically and is complemented by additional data.

The network discovery is based on MAC address information, being able to identify network device uplinks and end node connections by physical address. However, a pure network discovery does not reveal the nodes' function or geographical location, so external databases are queried to complete the topology view (e.g. ATLAS RackWizard, IT LanDB, Sysadmin ConfDB).

Self-check Mechanism

The reliability of the system may be affected by a misbehaving module. The purpose of the self-check mechanism is to identify and inform experts in real-time about any faulty modules. To make the self-check mechanism more robust, the checking of module execution status and the mail reporting have been implemented as two distinct components.

The execution status check is implemented using a set of scripts that monitor each module's functionality and expected results, and then report the status to the N-CDB. A module can also flag a warning and send a detailed message if it anticipates an imminent problem likely to require expert intervention. The warning component then checks all the active module status information and sends error-triggered and daily service status mail reports.



Statistics Collection

Gathering statistics about traffic flows and network state is mandatory for monitoring the system's performance. The network overview statistics are the primary source of information for investigating most network problems. These statistics are collected by APoll, an internally developed service, which polls SNMP counters from switches and routers every 30 seconds. Once a network issue is isolated, an in-depth analysis is performed by looking for relevant traffic samples collected via the sFlow protocol.

Sometimes external factors, such as environmental conditions or faulty systems, affect the network's normal behaviour. Making information from directly related systems available to network monitoring modules gives the networking team an advantage in understanding and limiting the impact of an external event. Currently, full or partial information from NAGIOS (computer monitoring), PVSS (environmental conditions database) and DAQ/IS (data-taking information service) is accessible through the same user interface used for network monitoring.

The User Interface

The user interface is designed to provide all the data an expert needs to investigate a problem and yet be simple to use for non experts. A set of web-based applications, integrated as a portal, offer access to real-time reports and historical statistics about network state and complementary systems. For offline visualization of the current topology a comprehensive PDF report is generated daily to be downloaded.

Another option to access the N-CDB data is through a CLI-like interface. The CLI tool was developed to allow experts to access and change the N-CDB data manually without the risk of affecting its low level consistency.

The ADAM API

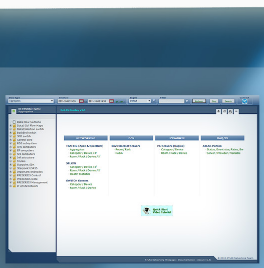
The information stored in the N-CDB is also of interest to shifters and system analysts. To facilitate programmatic access to information, a generic interface for data exchange has been implemented. The creation of a generic interface has been a joint effort between several ATLAS teams and lead to the definition of the ADAM (ATLAS Data API Mechanism) data exchange interface.

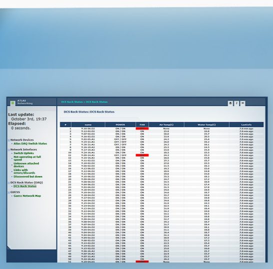
The ADAM API is fully implemented in the network monitoring solution and provides network traffic information, computer statistics and environmental conditions to external applications on demand. Through this API the network monitoring software can be a data provider for other external data analysis applications.

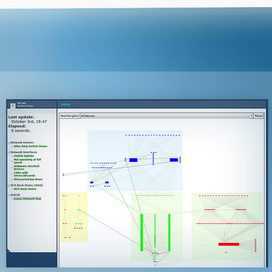
Current status & future plans

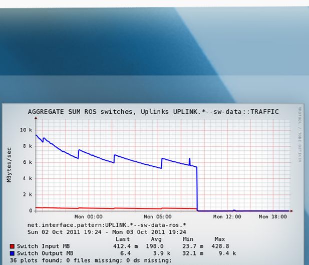
Started as an independent network monitoring software, the current ATLAS DAQ network monitoring solution has grown by integrating additional system information. This extension improved the understanding of overall system behaviour by enabling the correlation between networking events and external factors. Two years after its deployment it is the main tool used by both experts and non experts to analyze network problems. It has proven to be better suited for system diagnostics than the alternative commercial solution used for performance monitoring.

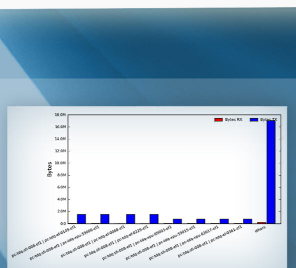
Future plans to improve the current solution include the extension of the N-CDB by storing network event logs from multiple sources and the addition of an event processing engine. Such a feature will bring together the network events signaling a state change, with the collected statistics that provide a more complete picture of the circumstances associated with the generated event.

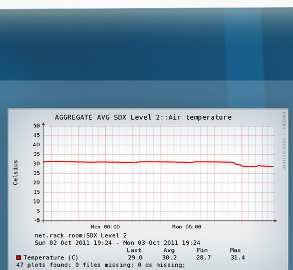







AGGREGATE SUM ROS switches, Uplinks UPLINK.*--sw-data::TRAFFIC
MBytes/sec
10 k
8 k
6 k
4 k
2 k
0
Mon 00:00
Mon 06:00
Mon 12:00
Mon 18:00
net.interface.pattern:UPLINK.*--sw-data-ros.*
Sun 02 Oct 2011 19:24 - Mon 03 Oct 2011 19:24
Last Avg Min Max
Switch Input MB 412.4 m 198.0 23.7 m 428.8
Switch Output MB 6.4 3.9 k 32.1 m 9.4 k
36 plots found; 0 files missing; 0 ds missing;


Bytes RX
Bytes TX
Bytes
18.0M
16.0M
14.0M
12.0M
10.0M
8.0M
6.0M
4.0M
2.0M
others


AGGREGATE AVG SDX Level 2::Air temperature
Celsius
50
45
40
35
30
25
20
15
10
5
-0
Mon 00:00
Mon 06:00
net.rack.room:SDX Level 2
Sun 02 Oct 2011 19:24 - Mon 03 Oct 2011 19:24
Last Avg Min Max
Temperature (C) 29.0 30.2 28.7 31.4
47 plots found; 0 files missing; 0 ds missing;