Journal of Physics: Conference Series **513** (2014) 062046 doi:10.1088/1742-6596/513/6/062046

System performance monitoring of the ALICE Data Acquisition System with Zabbix

A. Telesca[1], F. Carena[1], W. Carena[1], S. Chapeland[1], V. Chibante Barroso[1], F. Costa[1], E. Dénes[2], R. Divià[1], U. Fuchs[1], A. Grigore[1,3], C. Ionita[1], C. Delort[1], G. Simonetti[1,4], C. Soós[1], P. Vande Vyvre[1], and B. von Haller[1] for the ALICE collaboration

[1] European Organization for Nuclear Research (CERN), Geneva, Switzerland
[2] KFKI Research Institute for Particle and Nuclear Physics, Wigner Research Center, Budapest, Hungary
[3] Politehnica Univesity of Bucharest, Bucharest, Romania
[4] Dipartimento Interateneo di Fisica M. Merlin , Bari, Italy

E-mail: `adriana.telesca@cern.ch`

Abstract. ALICE (A Large Ion Collider Experiment) is a heavy-ion detector studying the physics of strongly interacting matter and the quark-gluon plasma at the CERN LHC (Large Hadron Collider). The ALICE Data-AcQuisition (DAQ) system handles the data flow from the sub-detector electronics to the permanent data storage in the CERN computing center. The DAQ farm consists of about 1000 devices of many different types ranging from direct accessible machines to storage arrays and custom optical links. The system performance monitoring tool used during the LHC run 1 will be replaced by a new tool for run 2. This paper shows the results of an evaluation that has been conducted on six publicly available monitoring tools. The evaluation has been carried out by taking into account selection criteria such as scalability, flexibility, reliability as well as data collection methods and display. All the tools have been prototyped and evaluated according to those criteria. We will describe the considerations that have led to the selection of the Zabbix monitoring tool for the DAQ farm. The results of the tests conducted in the ALICE DAQ laboratory will be presented. In addition, the deployment of the software on the DAQ machines in terms of metrics collected and data collection methods will be described. We will illustrate how remote nodes are monitored with Zabbix by using SNMP-based agents and how DAQ specific metrics are retrieved and displayed. We will also show how the monitoring information is accessed and made available via the graphical user interface and how Zabbix communicates with the other DAQ online systems for notification and reporting.

1. Introduction

1.1. The ALICE experiment and its Data Acquisition system

ALICE (A Large Ion Collider Experiment) [1] is a general purpose, heavy-ion detector at the CERN LHC designed to investigate the physics of strongly interacting matter and the quark-gluon plasma in nucleus-nucleus collisions. It consists of 18 different detectors and five main online systems of which the Data Acquisition is part. The ALICE Data Acquisition system [2] is in charge of handling the complete data flow from the sub-detector read out electronics to the archiving on tapes in the CERN Computer Center. Data fragments coming from optical links (Detector Data Links DDL) connected to the sub-detector electronics are aggregated by a first level of computers. Successively, data are sent to a second stage where they are processed, formatted and recorded to disk arrays before completing the migration to the permanent storage.

1.2. The ALICE DAQ farm

These tasks are performed by nodes installed in the ALICE DAQ farm, which consists of about 1000 devices of different types. The system monitoring of the ALICE DAQ farm has been performed by Lemon [3] during Run 1. Given that the long-term Lemon support provided by CERN IT was becoming uncertain and new tools with interesting functionalities were becoming available, we decided to investigate options different from Lemon for the ALICE DAQ system monitoring.

The ALICE DAQ nodes are categorized according to the roles they play in the data flow. The LDCs (Local Data Concentrators) are in charge of aggregating data fragments into subevents. The GDCs (Global Data Collectors) perform the complete event building and are connected to a Storage Area Network (SAN) for the transfer of data to the Transient Data Storage (TDS), which consists of disk arrays used for the temporary storage. The DQM (Data Quality monitoring) and the DA (Detector Algorithm) machines are respectively in charge of the data quality monitoring and calibration tasks for the detectors. In addition, there are infrastructure devices such as network switches, power distribution units (PDU), database and web servers, and other DAQ-related roles.

The DAQ farm devices can also be classified into two big groups: the nodes that can be directly accessed and allow scripts execution (i.e. Linux hosts in charge of the data aggregation, recording, data quality monitoring, etc) and the remote nodes that do not allow direct programmatic access (i.e. network routers, disk arrays and PDUs) and whose monitoring parameters can be retrieved through SNMP or command line interface.

The number of devices in the ALICE DAQ farm will increase following the two shutdown periods of the LHC and the associated experiment upgrades and it is estimated to reach 2000 nodes in 2018.

2. Monitoring system characteristics and needs

The specificities of the DAQ farm previously described in terms of number of devices and different roles translate into a list of characteristics that the monitoring system to be used during the run starting in 2015 should have. The new system monitoring tool must:

- Collect information in a light and efficient way ensuring a low impact on the system performance.
- Collect information from both directly accessible (i.e. via monitoring client, script, etc) and remotely accessible devices (i.e. via SNMP).
- Scale well to at least 2000 nodes.
- Be extensible, customizable and give the possibility to easily access the stored data. It must provide an interface where experts and shifters can monitor the DAQ farm status.
- Interface with other DAQ software packages and devices such as the DAQ central reporting system (Orthos [4]) for alarming and reporting.

The parameters that the monitoring system should be able to retrieve can be either general or specific to the role that a given device plays in the dataflow. For Linux devices, general metrics can include CPU utilization, memory and partition statistics as well as network traffic while for specific roles such as LDC, we are interested in monitoring, for instance, the DDL in/out rate the CPU and memory utilization of particular processes. The DDL rate, for instance, should be retrieved on the machine via a dedicated script. For remote devices, such as PDUs, most of the times we are limited to what the SNMP interface provides.

3. Comparison of different tools

The new monitoring tool was selected following an evaluation and comparison of different tools. A first shortlist has been created by focusing on tools that have free software licenses and are open source. Among the ones shortlisted, we have considered the tools with: SNMP capabilities, alerts/triggers, distributed monitoring, logical grouping and a large user community. This has resulted in four main tools that have been included in the preliminary analysis: Icinga, Cacti, Zenoss and Zabbix. Another tool, called Splunk, has been considered because, at the time of the evaluation, CERN-IT was evaluating it for their own purposes. The MonALISA monitoring tool that has been specifically developed for CERN (and wider) usage has also been taken into account.

3.1. Icinga

Icinga [5] is a system monitoring tool providing information on the monitored system health. As it is mainly focused on status checks and notifications, its tasks overlap with the main role of Orthos. Icinga does not provide a native interface for performance data monitoring. In order to have a flexible performance data display, additional tools would need to be installed.

3.2. Cacti

Cacti [6] is a monitoring tool designed to provide graphing functionality and performance data retrieval. It is a poll monitor system and privileges SNMP as the main way of data retrieval. However, during several tests in the DAQ lab, some graphs showed missing data although no errors could be found in the Cacti logs. The cause could be some connection problems, when, for instance, the machine does not respond to an SNMP request from the server.

3.3. Zenoss

Zenoss [8] is an agent-less monitoring tool that provides a user interface from which users can monitor and configure the system. It is very easy to install and monitor standard parameters on many machines. However, its extension is less straightforward than expected and debugging was found to be difficult.

3.4. Zabbix

Zabbix [7] is a very flexible monitoring tool which allows both polling and trapping. It can be used either through a server-agent application where the agent sends data to the server or through one or more servers that poll the results from the monitored devices. Zabbix works very well and is scalable when using its agents running on the monitored host and distributed monitoring.

3.5. Splunk

Splunk [9] is a very flexible tool to search, monitor and analyse machine-generated data. It can be used to retrieve system metrics from hosts and remote devices as well as to troubleshoot problems. It also provides a very efficient search over indexed machine data that allows the user to aggregate and correlate information.

3.6. MonALISA

MonALISA [10] is an efficient software that is able to provide complete monitoring for complex systems. It has been mainly designed to monitor Grid nodes and to exchange information and processing tasks among them. As it has mainly been conceived to serve a highly distributed system, the usage of MonALISA standalone (locally) would represent a limited version of its whole functionality.

3.7. Comparison and decision

All the tools analysed provide alerting, distributed monitoring, access control and a web interface allowing the user to display the metrics and configure the monitoring tool. The following tables summarize other important characteristics of the tools described above.

Table 1. Monitoring tools comparison

(a)

Name	Data gathering	Graphing (0-2)	Triggering (0-1)	Scalability (0-2) #hosts	Data Storage	Extensibility (0-2)
Icinga	Agent	0	1	1 - up to 1000	DB	2
Cacti	Server	2	0	1 - up to 1000	RRDtool - DB	2
Zenoss	Server	1	1	2 - 1000+	RRDtool - DB	1
Zabbix	Agent or Server	2	1	2 - 1000+	DB	2
Splunk	Agent	2	1	2 - 1000+	raw files	2
MonALISA	Agent	2	1	2 - 1000+	DB	2

(b)

Name	SNMP (0-2)	Documentation User Community (0-2)	Max Granularity (0-2)	Auto Discovery (0-2)	Free (0-1)	**Total**
Icinga	2	2	1-1 minute/metric	2	1	**12**
Cacti	2	2	1-1 minute/metric	1	1	**12**
Zenoss	1	1	1-1 minute/collector	2	1	**11**
Zabbix	2	2	2-No limit/metric	2	1	**16**
Splunk	2	2	2-No limit/metric	2	0	**15**
MonALISA	2	1	1-1 minute/metric	2	1	**14**

Where:

- Data gathering is the way in which monitored data are gathered.
- Graphing refers to the native graphing functionality of the tool. Icinga does not provide a graphing functionality unless an external tool is installed. Cacti, Zabbix, Splunk and MonALISA provide graphing functionality that allows the user to customize the graphs, aggregate and modify them. Zenoss provides graphing but it does not allow for flexible customization and zooming.
- Scalability refers to the maximum number of hosts that can be efficiently monitored.
- Flexibility/Extensibility has been rated from 0 to 2 indicating how easy it is to extend the tool in terms of metrics and graphs. Zenoss is less easy to customize than the other tools under consideration.
- SNMP represents the capability to monitor devices using the SNMP protocol. With Zenoss, it is more complex than with others tools to add SNMP based metrics. With Splunk SNMP metrics can be obtained by querying devices via scripts.
- Documentation/User community refers to how easy it is to find information or problem resolution from community resources (i.e. forum, wiki, etc).
- Maximum granularity represents the minimum check interval. For Icinga, Cacti and MonALISA, it can be set to a minimum of 1 minute, while there is no limit for Zabbix and Splunk. All these tools allow for setting the check interval by parameter. For Zenoss, it is set at the poller level and all the metrics use the same polling interval.
- The auto discovery allows for discovering devices automatically, which can be very useful for large deployments. In Cacti and Icinga, auto discovery can be installed through a plugin while in Zenoss, Zabbix and MonALISA it is natively integrated in the tool. For Splunk, the device becomes known to the server once the Splunk agent is installed and running. For Cacti, the auto-discovery was less efficient than for the other tools.

- The parameter Free refers to the cost to purchase/use the tool. Splunk is a licensed tool.

The total is the sum of the mark given to each parameter. This gives only an indication as every single parameter should be discussed according to the needs. Naturally, the scores have been given considering the specific needs of the ALICE DAQ monitoring system.

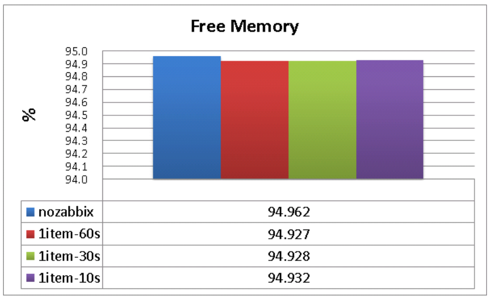
The considerations above made us select Zabbix for further testing and as a candidate for the system monitoring of the ALICE DAQ farm.

4. Zabbix

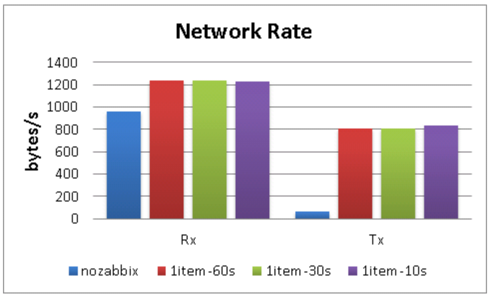
Zabbix has been installed in the ALICE DAQ lab in order to evaluate it more in details in terms of functionality and impact on the system performance. The ALICE DAQ lab, of which 70 devices have been used for the tests, represents a good environment to assess the suitability of Zabbix to monitor the ALICE DAQ farm in production during the 2015 run.

4.1. Tests on impact on system performance

Extensive tests have been performed in order to understand the impact of the Zabbix client processes on the system performance. Figure 1(a) and Figure 1(b) respectively show how the free memory and network rate of a host change as a function of whether and at what collecting frequency Zabbix is running. The figures show that the introduction of Zabbix in the system



(a) Free memory



(b) Rate Rx/Tx

Figure 1. System resource usage with and without Zabbix

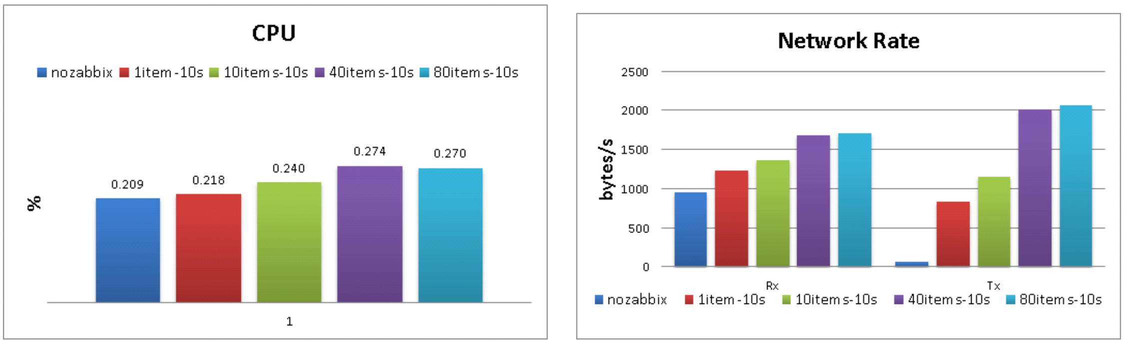
causes a decrease of the free memory and of the network rate. However, in the worst cases, the free memory decreases by 0.035% while the rate increases by 800 bytes/s in output and by 300 bytes/s in input. This is very acceptable in view of the foreseen use case.

If we compare the system performance for different configurations of the Zabbix client where we change the number of values retrieved and we keep the values retrieval interval fixed, we observe similar results. The results are shown in Figure 2(a) and Figure 2(b).

We can see here a more realistic scenario where the number of values retrieved by Zabbix is 80. The expected configuration of device roles such as LDC, GDC, DQM and DA will be of about 80 values with mixed retrieval intervals from 10 seconds to 60 seconds. Figures 2(a) and 2(b) show that the maximum reached network rate is 1700 bytes/s in input and about 2 kbytes/s in output while the used CPU reaches a maximum of 0.27%. All results represents acceptable system performance.

4.2. Zabbix usage in the ALICE DAQ farm

Zabbix clients have been configured to retrieve different metrics according to which role the device plays in the ALICE data flow. The Zabbix dashboard provides functionality to create



(a) CPU idle (b) Rate Rx/Tx

Figure 2. System resource usage without Zabbix and with different number of values retrieved

maps and tables that give a good overview of the system status. The main entry point of the ALICE Zabbix monitoring web interface is the map shown in Figure 3.

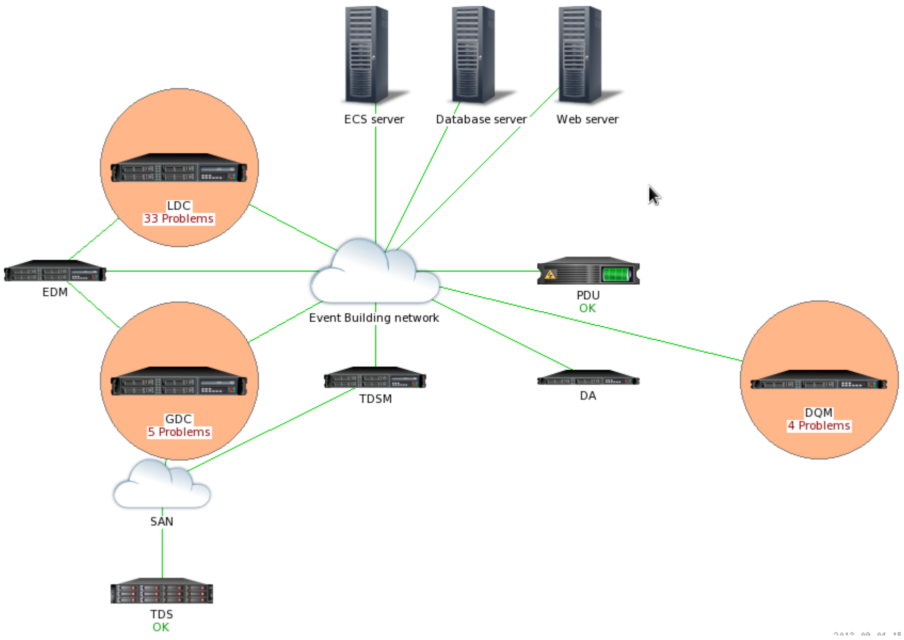


Figure 3. Zabbix ALICE DAQ map

From the map it is possible to recognize the roles that have problems (circled in orange) and the ones that are in a good status. This allows the user to easily spot problems. In addition, the map represents the main point from which the user can access the details about all the device roles such as role graphs as shown in Figure 4. In addition data overviews and information about alarms have been created; they are shown in Figures 5(a) and 5(b), respectively.

5. Conclusions

The system performance monitoring tool used by the ALICE DAQ during the LHC run 1 will be replaced by a new tool for the run starting in 2015. An evaluation of different available monitoring tools has been carried out and resulted in the selection of Zabbix as a good candidate

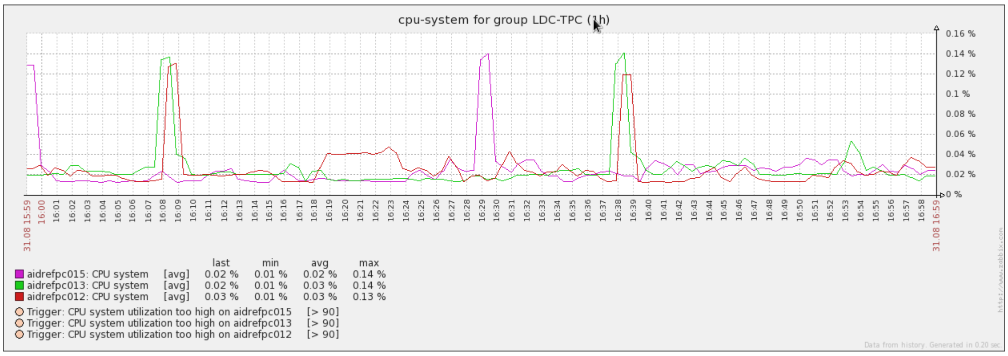


Figure 4. LDC CPU usage

Hosts	Air flow dew point	Air flow flow	Air flow humidity	Air flow temperature	DeciAmps outlet 1	DeciAmps outlet 2	DeciAmps outlet 3	DeciAmps outlet 4	DeciAmps outlet 5	DeciAmps outlet 6	DeciAmps outlet 7	DeciAmps outlet 8	DeciAmps outlet 9	Feedback outlet 1	Feedback outlet 2
phaidpdu01	8	29	53 %	17 C	1 A	1 A	1 A	1 A	1 A	1 A	1 A	1 A	1 A	OK (1)	OK (1)
phaidpdu03	8	25	51 %	18 C	1 A	1 A	1 A	1 A	1 A	1 A	1 A	1 A	1 A	OK (1)	OK (1)

(a) Data overview

System status

Host group	Disaster	High	Average	Warning	Information	Not classified
DQM	0	0	3	1	0	0
GDC	0	0	5	0	0	0
LDC	0	0	33	0	0	0
MON	0	0	6	0	0	0
PDU	0	0	0	0	0	0
TDS	0	0	0	0	0	0
Zabbix servers	0	0	1	1	0	0

Updated: 15:38:19

(b) Zabbix system status

Figure 5. Zabbix data overview and system status

for the system performance and exception monitoring of the ALICE DAQ farm. Extensive tests performed in order to assess the impact of Zabbix on the system performance demonstrated that the overhead introduced by Zabbix is low and therefore acceptable. Zabbix has been installed in the ALICE DAQ lab to retrieve metrics at different intervals according to the various machine roles in the ALICE data flow. The Zabbix graphical interface has been configured to act as the main entry point for the users to access all information related to the system health and performance in order to easily spot problems and check the system behaviour. So far, the experience with Zabbix is very positive. The Zabbix monitoring tool will be installed in the production system during the LHC long shutdown of 2013-2014 and will be ready for the data taking of Run 2 in 2015-2017.

References

[1] ALICE Collaboration, *The ALICE experiment at the CERN LHC*, 2008 JINST **3** S08002, 2008.
[2] ALICE Collaboration, *ALICE technical design report of the trigger, data-Acquisition, high level trigger, and control system*, CERNLHCC- 2003-062, 2004.
[3] https://lemon.web.cern.ch/
[4] S. Chapeland et al., *Orthos, an alarm system for the ALICE DAQ operations*, Proc. Computing in High Energy and Nuclear Physics 2012, New York, NY, USA, 21 - 25 May 2012, pp.012013
[5] https://www.icinga.org/
[6] http://www.cacti.net/
[7] http://www.zabbix.com/
[8] http://www.zenoss.com/
[9] http://www.splunk.com/
[10] http://monalisa.caltech.edu/