Monitoring and Analytics at INFN Tier-1: the next step

Fabio Viola[1,*], *Barbara* Martelli[1,**], *Diego* Michelotto[1], *Enrico* Fattibene[1], *Antonio* Falabella[1], *Stefano* Dal Pra[3], *Lucia* Morganti[1], *Luca* Dell'Agnello[1], *Daniele* Bonacorsi[2,3], and *Simone* Rossi Tisbeni[1]

[1]INFN-CNAF, Bologna, Italy
[2]INFN, Bologna, Italy
[3]DIFA, University of Bologna, Bologna, Italy

Abstract. In modern data centres an effective and efficient monitoring system is a critical asset, yet a continuous concern for administrators. Since its birth, INFN Tier-1 data centre, hosted at CNAF, has used various monitoring tools all replaced, a few years ago, by a system common to all CNAF departments (based on Sensu, Influxdb, Grafana). Given the complexity of the inter-dependencies of the several services running at the data centre and the foreseen large increase of resources in the near future, a more powerful and versatile monitoring system is needed. This new monitoring system should be able to automatically correlate log files and metrics coming from heterogeneous sources and devices (including services, hardware and infrastructure) thus providing us with a suitable framework to implement a solution for the predictive analysis of the status of the whole environment. In particular, the possibility to correlate IT infrastructure monitoring information with the logs of running applications is of great relevance in order to be able to quickly find application failure root cause. At the same time, a modern, flexible and user-friendly analytics solution is needed in order to enable users, IT engineers and IT managers to extract valuable information from the different sources of collected data in a timely fashion. In this paper, a prototype of such a system, installed at the INFN Tier-1, is described with an assessment of the state and an evaluation of the resources needed for a fully production system. Technologies adopted, amount of foreseen data, target KPIs and production design are illustrated.

1 Introduction

Maintenance is the process of preserving or restoring the good operating conditions of a system. Different approaches to maintenance exist. Corrective maintenance is probably the most intuitive approach: when something breaks, it is fixed or replaced. It has the clear drawback of producing a downtime while the operating state is restored. Preventive (or periodic) maintenance, as the term suggests, aims at replacing components before a fault occurs. This is achieved through periodic scheduled interventions. Checks, revisions and replacements are planned based on the observations of the past events and on the estimated life of each component. This approach has the clear advantage of reducing failures and downtime by

---

*e-mail: fabio.viola@cnaf.infn.it
**e-mail: barbara.martelli@cnaf.infn.it

continuously keeping every component in good conditions. On the other hand, it is an expensive approach, since many hardware components could have a longer life and are instead replaced due to a scheduled intervention. This motivates the approach known as condition-based maintenance: the continuous monitoring of a set of measurements can easily highlight abnormal behaviors, allowing the administrator to perform a maintenance intervention before a fault occurs [1]. Similarly, also predictive maintenance exploits continuous observations of equipment. However, predictive maintenance has the aim to identify a temporal trend that could help to foresee when a maintenance intervention will be necessary. Predictive maintenance is currently applied in a wide set of application domains, starting from wind turbines [2-3] , to predictive car maintenance [2], or smart manufacturing [5-6] [3, 4] just to name a few.

In this paper we focus our attention on the data centres domain. In fact, we hereby present a layered scalable big data infrastructure that, upon completion, will be aimed at predictive maintenance for large data centres, going beyond the pure log-based predictive maintenance [5]. This research activity leverages previous works carried out at INFN-CNAF [8-10] . In order to perform predictive maintenance, it is important to collect all the available information of the resources to be monitored and provide it in an effective and efficient way to the devoted software that will carry out the analysis. In this paper we propose a chain of tools based on open source Apache technologies. Nonetheless, the proposed infrastructure (supporting both batch and real-time analysis) follows a general approach ensuring its portability to different frameworks. On the bottom level, Apache Flume performs data ingestion dealing with different data sources (e.g. syslog, time series databases). Timely distribution to processing nodes is carried out at a higher level by the topic-based publish-subscribe engine Apache Kafka. Processing is performed through Apache Spark and Spark Streaming instances. Data persistence is achieved through Apache's distributed filesystem HDFS, that allows saving both the original log files and the results of the analyses. The topmost layer constitutes the presentation layer, including the visualization tools through which the final users (i.e., sysadmins) may monitor the status of the system and be notified about foreseen faults. This work is framed in the context of the INFN Tier-1 data centre, involving data from approximately 1200 nodes. INFN-CNAF's Tier-1 site is the major Italian data centre in the Worldwide LHC Computing Grid[1] (WLCG). This data centre features 40,000 CPU cores, 40 PB of disk storage, 90 PB of tape storage, and it is connected to the Italian (GARR) [2] and European (GEANT) [3] research network infrastructure with more than 200 Gbps. In such a scenario, adopting a predictive maintenance approach would allow INFN saving costs deriving from downtime (thus providing a better service) or unnecessary hardware replacement. Moreover, a maintenance intervention performed before a fault occurs, may allow avoiding losses of data.

In Section 2, the proposed infrastructure is described from a high-level perspective. Section 3 focuses on the discussion of the envisioned data sources. Section 4 describes in detail the main component of the infrastructure: the big data cluster. Finally, in Section 5, conclusions are drawn.

2 The predictive maintenance infrastructure at a glance

To achieve the final goal of an infrastructure for predictive maintenance of the INFN's Tier-1 data centre, the first step is to identify all the data sources in the system. Then, all the information collected should be conveyed to a big data analysis cluster, in order to be continuously

---

[1]http:/wlcg.web.cern.ch/
[2]https://www.garr.it/
[3]https://www.geant.org/

processed. This high-level simplified view is depicted in Fig. 1: on the left, data sources (better detailed in Section 3) publish their data on a centralised collector ("collog" in Fig.1). This feeds the Big Data Cluster ("BDC" in Fig.1) dedicated to the ingestion and analysis of the available information.
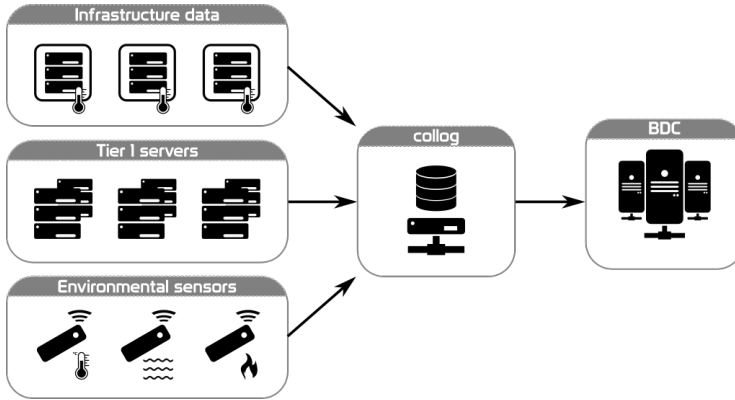
Figure 1. High-level view of INFN's Tier-1 big data infrastrucure for log analysis

After a more detailed description of the involved data sources, every component of the big data cluster will be introduced.

3 Data sources

In the complex scenario of the INFN's Tier-1 data centre, data sources can be mainly ascribed to the following three categories:

Log files – This category embraces all the log files produced by the machines hosted in the data centre. Log files may refer to system processes and services (e.g. cron, sudo, sshd) as well as log files produced by the HEP experiments running on the nodes. Volume of these files ranges from approximately 7.5 GB/day for days with no particular events, to about 25 GB/day in days with an intensive use of the data centre.

Sensor networks – The rooms which currently host the data centre will be equipped with sensor networks to collect data related (among others) to temperature, humidity and flooding or fire events.

Racks and cabinets – Racks and cabinets hosting all the computing, networking and storage modules already embed sensors (e.g. temperature and humidity) gathering useful information. This information allows a fine-grained monitoring.

4 Software Components of the Big Data Cluster

4.1 Apache Flume

The high level view of the CNAF Big Data Cluster is depicted in Fig. 2.From the ground up, the first layer of our big data platform is the data ingestion tool, in this case represented by Apache Flume[4]. Flume is designed to efficiently handle large amounts of data coming from
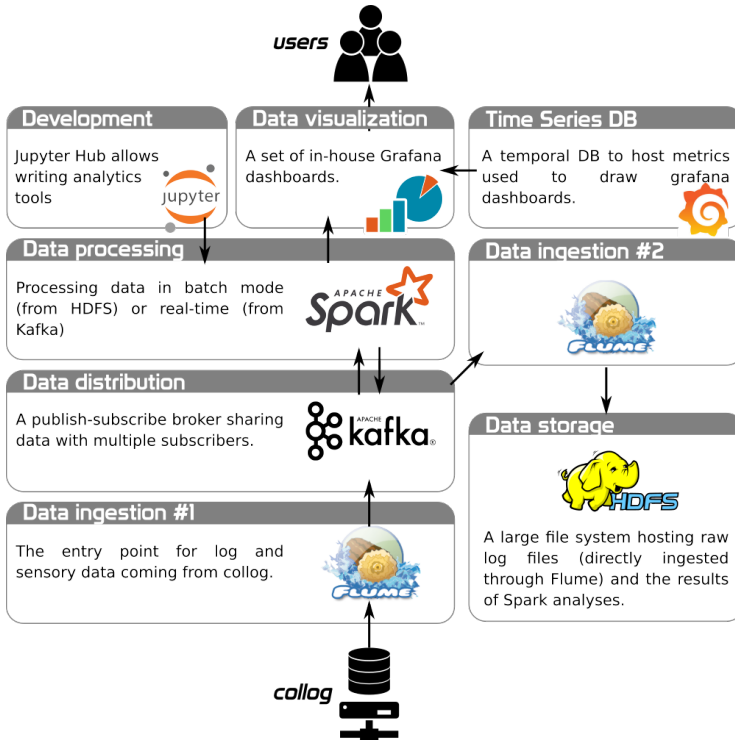
---

[4]https://flume.apache.org/

Figure 2. Detailed view of the Big Data Cluster

heterogeneous sources and flowing to heterogeneous recipients. More in particular, Flume is able to retrieve (and write) data from (to): Avro, Thrift, Exec, JMS, Spool directories, Taildir, Twitter, Kafka, Netcat, and many other. This makes Flume suitable for the needs of INFN's Tier-1 data centre. Flume applications rely on three types of components: sources, channels and sinks. While sources have already been discussed in Section 3, channels and especially sinks deserves a further explanation. Sinks are the recipients of the data ingestion process. In the presented big data platform, the recipient is Kafka, that will be described in the following subsection. The channel instead, is a memory buffer interconnecting sources to sinks.

4.2 Apache Kafka

Apache Kafka[5] is a topic-based publish-subscribe [6] data distribution system. In the proposed architecture, the role of Apache Kafka is to decouple the data ingestion tool from the consumers, allowing in this way to broadcast data to multiple subscribers. In the proposed architecturs, a set of eight topics has been adopted. Topics could be considered as four couples of raw and enriched logs, each belonging to a different division of the data centre. Since Kafka represents a fundamental building block of the platform, it is important to grant fault tolerance and high performance. Both have been achieved through a set of three brokers with two replicas for each of the three partitions of a topic. Retention has been set to three days.

---

[5]https://kafka.apache.org/

4.3 Apache Spark

Apache Spark[6] is the main tool through which data analysis is performed on the cluster. Users are allowed to choose between batch and streaming analysis. In the first case, they may rely on data stored on the HDFS file system or retrieve data through Kafka (but in this case the available data is limited by the retention period). Real-time analysis tools subscribe to one or more Kafka topics and process data as soon as they arrive. In both cases, results may be directly stored on the HDFS file system, or streamed again through Kafka (on the topics dedicated to enriched data).

4.4 Apache HDFS

HDFS[7] (Hadoop Distributed File System) is Apache's solution to distributed file storage. In the proposed architecture, a cluster of HDFS nodes is used to provide a fault tolerant layer to host both raw files and enriched files (i.e. those resulting from a Spark analytics process). The process of copying data from Kafka to Apache HDFS is carried out by a second instance of Flume where the source is of course Kafka and the sink is HDFS.

4.5 Jupyter Hub

The complex infrastructures to ingest and process log files and sensory data requires an easy user interface in order to be adopted by the INFN's team of data scientists. Jupyter Hub[8] meets this requirement, providing a web interface where users may access online notebooks to program with Scala or Python and the Spark framework.

4.6 InfluxDB

The big data cluster presented in this paper also embeds a time series database, in order to collect and monitor the temporal evolution of a set of metrics. In the specific case, the time series database is InfluxDB[9] in its open source edition. Data is generated either by end users through Spark or through a set of bash scripts developed by sysadmins. In the first case, the monitored metrics pertain the log and sensory data analysis, while in the second the focus is the analysis of the state of the system.

4.7 Grafana

In such a complex architecture, where the volume of data is too high to allow a fine-grain analysis of each of the collected information by hand, a visualization system plays a crucial role. Grafana[10] provides a client-server application to easily develop effective dashboards allowing users to monitor the metrics of interests. Fig. 3 shows a screenshot of the dashboard built to monitor the status of the Big Data Cluster with respect to a subset of the available nodes (i.e. those running INFN's service StoRM).
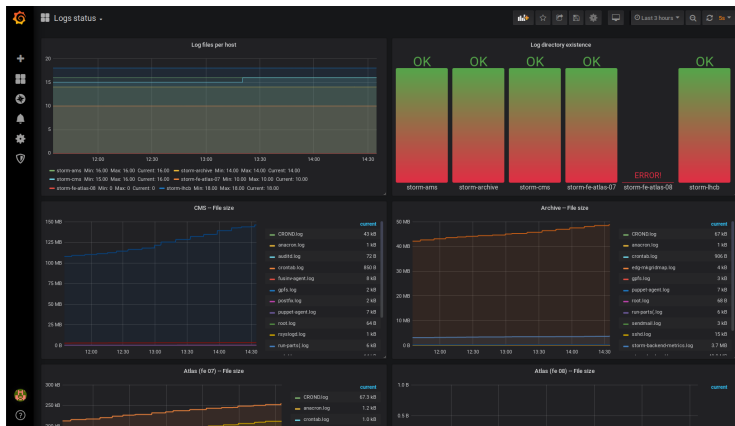
---

[6]https://spark.apache.org/
[7]https://hadoop.apache.org
[8]https://jupyter.org/hub
[9]https://www.influxdata.com/
[10]https://grafana.com/

Figure 3. Grafana dashboard to monitor the status of the Big Data Cluster

4.8 Infrastructure setup and deployment

DODAS (Dynamic On Demand Analysis Service) [7] has been adopted to deploy and easily replicate the deployment of the analysis cluster. DODAS is a Platform as a Service (PaaS) tool allowing local and remote deployment with a minimal effort, based on the specifications included in TOSCA templates. It supports any cloud provider, only requiring the access credentials. Within DODAS, the Infrastructure Manager (IM) provides an abstraction over the underlying architecture.

5 Conclusion

In this paper, an infrastructure aimed at applying a predictive maintenance approach to a large data centre has been presented. This infrastructure represents a further evolution of the previous existing monitoring system developed at INFN-CNAF. In this scenario, new groundbreaking technologies, mainly from the Apache world, have been integrated. This grants the ability to collect data from multiple sources and allow users to 1) perform customised analyses using Spark via the comfortable user interface exposed by Jupyter; 2) inspect the status of the system through personalised Grafana dashboards.

Future works planned for this research activity involve several tasks. First of all, novel ingestion technologies (e.g. Fluentd) will be studied in order to replace Apache Flume, whose last stable version dates back to January 2019. Second, containers will be adopted for enduser applications, to enhance the security level of the system. A further future enhancement of the system envisions the integration of the Elasticsearch-Logstash-Kibana (ELK) stack in the presented architecture. As discussed in Section 3, an environmental sensor network will feed the system providing additional data on which to determine interesting insights. The completion of the platform with the above mentioned future works will finally allow the implementation of predictive maintenance tools based on advanced machine learning methodologies.

References

[1] M. Bevilacqua, M. Braglia, Reliability Engineering & System Safety **70**, 71 (2000)

[2] R. Dhall, V. Solanki, International Journal of Interactive Multimedia & Artificial Intelligence **4** (2017)

[3] K. Efthymiou, N. Papakostas, D. Mourtzis, G. Chryssolouris, Procedia CIRP **3**, 221 (2012)

[4] J. Wang, L. Zhang, L. Duan, R.X. Gao, Journal of Intelligent Manufacturing **28**, 1125 (2017)

[5] R. Sipos, D. Fradkin, F. Moerchen, Z. Wang, *Log-based predictive maintenance*, in *Proceedings of the 20th ACM SIGKDD international conference on knowledge discovery and data mining* (2014), pp. 1867–1876

[6] P.T. Eugster, P.A. Felber, R. Guerraoui, A.M. Kermarrec, ACM computing surveys (CSUR) **35**, 114 (2003)

[7] D. Spiga, M. Antonacci, T. Boccali, D. Ciangottini, A. Costantini, G. Donvito, C. Duma, M. Duranti, V. Formato, L. Gaido et al., *DODAS: How to effectively exploit heterogeneous clouds for scientific computations*, in *International Symposium on Grids and Clouds 2018* (2018)