lectito

Visualization System for Monitoring Data Management Systems

Emanuel Pinho[1], Alexandre Carvalho[1]

[1]INESC TEC, PORTUGAL

*Corresponding Author   alexandre.carvalho@inesctec.pt

ABSTRACT

Usually, a Big Data system has a monitoring system for performance evaluation and error prevention. There are some disadvantages in the way that these tools display the information and its targeted approach to physical components. The main goal is to study visual and interactive mechanisms that allow the representation of monitoring data in grid computing environments, providing the end-user information, which can contribute objectively to the system analysis. This paper is an extension of the paper presented at (Pinho and Carvalho 2016) and has the purpose to present the state of the art, carries out the proposed solution and present the achieved goals.

Keywords: Grid computing,Monitoring,Human-machine interaction, Graphical,Visual

WORLD'S DATA GENERATED

Nowadays, the amount of information electronically generated causes a constantly growing of database systems at (James 2015). The information we produce every day is huge. According to McAffe et al. (McAfee, et al. 2012), in 2012, about 2.5 Exabytes of data were daily created. Data is used by major technology companies as the main business strategy, performing their analysis and treatment to contribute for the perception of its customers' needs, as related by Mark Zuckerberg at (Zuckerberg 2015). Data is processed in complex, distributed and parallel systems, in order to improve their performance (Chen, Mao and Liu 2014). However, the effects of this overload usage can be an inconvenience: according to Andreozzi et al. (Andreozzi, et al. 2005), distributed systems require the existence of resources and services coordination, bringing the consequent change of decisions and behaviours. Such variations have repercussion effects on the performance of each component, making mandatory, in their maintenance process, the analysis of their performance. Therefore, a common approach has been to couple these systems with a monitoring system.

MONITORING SYSTEMS

Monitoring Systems are essential for any organizations having information systems performing critical tasks. Companies such as MongoDB (MongoDB 2015) characterize monitoring as a critical component in all of the database management processes. Xu (Xu 2011) relates that the handling of potentially heterogeneous systems requires tools able to capture information regarding real time performance. Sreeja *et al.* (Sreeja and Chaudhari 2015) claim that the failure detection through monitoring tools may help in recovery scenarios.

Usually, monitoring systems tasks are grouped by data collection, data processing and storage and data visualization. The first tasks of any system, which will be able to supply formatted data, is to ensure the access of this data. Therefore, the data collection made by sensors is the first task to perform, which can be centralized, distributed or even local. The second step focuses on store and process raw monitored data. The third stage is about visualization of accurate and easy to understand graphical representation of such data. Coherent with Silva and Catarci in (Silva and Catarci 2000), data captured by monitoring systems, representing performance values, mostly fall into time-evolving linear (regarding structure) data, and is usually represented by line graphs. However, other graphical methods can be used to represent such data.

PROBLEM

This research focuses on the last stage of a monitoring system: monitored data visualization. Currently, there are tools which have outstanding performance in collecting, storing and processing monitored data. However, currently available solutions for monitored data visualization still have room for improvement.

If the goal is about quickly identifying and preventing issues which may cause crashes, differentiation is an essential aspect: the system must be able to highlight information that deserves the user's attention. Furthermore, scalability is another problem in a complex system and can be a handicap to the schematic representation of a system being monitored. Its visual representation should remain clear, objective, unambiguous and intuitive. Therefore, usability must be preserved, not having an effect on the way that the user search and identify target components.

MONITORING TOOLS

This section presents the most relevant studied monitoring tools, their strengths and weaknesses.

A.   Ganglia (Ganglia 2001) is defined as a scalable and distributed tool used in systems with high performance, such as clusters and grids. According Andreozzi et al. (Andreozzi, et al. 2005), a complete ganglia system has three components: Gmond, Gmetad, Gweb. Gmond is responsible for the acquisition of values through sensors, which communicate with the operating system. Gmetad is the component that links Gmond to RRD database. According to Massie *et al.* (Massie, et al. 2012), the DRR files provide a dynamic allocation into different time segments, keeping constant database size. They claim that it is one of the most frequently used temporary data storage methods. The third component, and the one that have most relevant to this study, is the Gweb. It provides visual graphics to represent the data stored in the RRD files. According to Massie *et al.* in (Massie, Chun and Culler 2004), the Ganglia Web Client uses RRDTOOL to get useful information in PNG format through RRD files. The authors also report that the relationship between these components, designed hierarchically, is the biggest advantage over other tools.

B.   Observium (Observium Docs 2016) is defined as a monitoring tool able to discover their environment automatically. Thus, it is able to adapt itself to the system, using SNMP. Chris Blake says that this can be an advantage for users who do not want to spend much time with configurations. Although, it has more limitations, such as the availability to analyse extra metrics (Blake 2013). As Ganglia, Observium uses RRD file format for storing the values collected by the sensors.

C.   Nagios is another open-source tool, which has, according to (Nagios 2016), normal monitoring features, such as network services, capture capabilities from different machines, system alerts and graphical representation. However, as mentioned by Andreozzi *et al.* (Andreozzi, et al. 2005), its biggest problem is to have a little actions range, working only in local networks.

D.   Zabbix is able to monitor a complex distributed system, like Ganglia but with automatic mechanisms of network construction. The most relevant features are: the automatic protocols which they collect data, such as http, ftp, ssh, pop3, smtp, etc.; authentication and rules mechanisms; customized notifications; audit logs. However, as claim by Joseph (Joseph 2015), as the system is more complex in terms of visual organization, more complex can its usage be. Zabbix has also a poor configuration mechanism, which only can be made through xml files (Simmonds and Harrington 2009).

VISUALIZATION TECHNIQUES

Comparing a monitoring system with a story, (Tufte and Graves-Morris 1983) claim that the main goal of both is to explain something to someone. As a communication process it is mandatory that the emitter uses the most appropriate "language" that the receiver is able to understand. If a monitoring system is an artefact supporting a communication process, the designer of communication process must first realize what is the most suitable visual metaphor for the representation of monitored data, and to do that, has to understand, among other data structure (linear, hierarchical, etc.), dimensions, number of dependent variables, how can they change, etc. Graphic variables such as colour, texture, edges, transparency are some to encode visual information. For instance Healey (Healey 1996) identifies the color as an important visual feature and often used to differentiate details. Other concerns about interaction and layout matter: Buja *et al* (Buja, et al. 1991) and Mackinlay *et al*. (Mackinlay, Card and Robertson 1990) present zooming and layout manipulation as good assets to define points of interest. Diagram representation is, according to liinsky and Steele (Iliinsky and Steele 2011) and excellent technique to focus attention.

PROPOSED SOLUTION

After having identified the strengths and weaknesses of the studied monitoring tools we concluded that Ganglia performs very well regarding monitoring data capture and storage when applied to complex and distributed systems. Therefore, Ganglia was selected as the grounding roots for the developed testing workbench. Our solution is built over a Ganglia system using Gmond and Gmetad components. A Graphite instance is used to bridge between the RRD data storage and the web client application where we tested our improvements. The proposed solution relies on numerical data being transferred from the server and on the fact that charts are built on the client-side. The graphical processing is supported on D3.js (D3 Data-Driven Documents 2016) technology, which is not a graphical building tool, but provides DOM manipulation that allows the construction of any data representation. Tests were conducted on cluster of three servers where one of the servers hosts a Gmetad instance. This server (Host 1) also hosts an instance of graphite, which communicates with the client web application. The structure is presented in 0
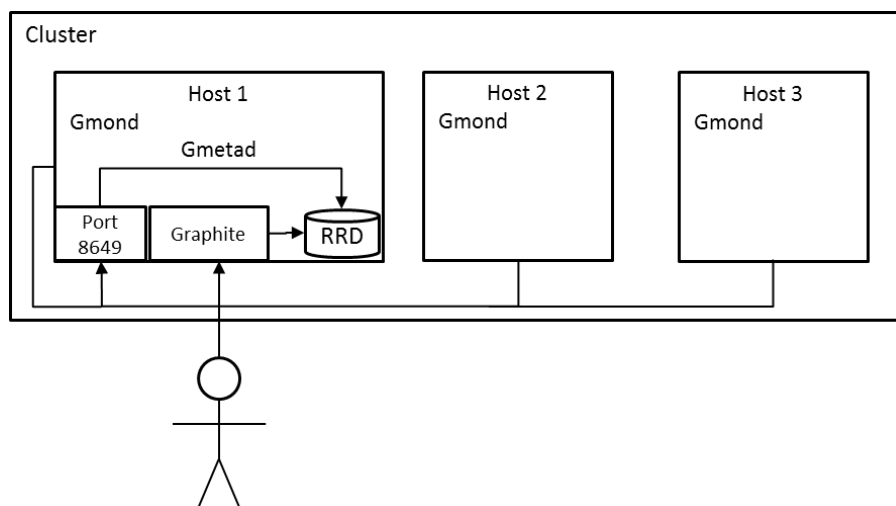

Figure 1. System physical architecture

Focusing on the monitor data visualization, we experimented visualization techniques that can contribute to a more easy perception and focus on the user's attention, most especially when the system scales and where the search for components may become a hard task. We propose the use of the following charts to help facilitate the user to navigate through the monitored components and to select components for further inspection of corresponding metrics dashboards.

- Tree charts are good to represent hierarchical data: most of the surveyed tools focus on the graphical representation of the physical organization of the system being monitored. The structure representation is a key point to understand and easily identify components. For instance, Ganglia uses breadcrumbs technique to help the user move through grids, servers and metrics. However, through this organization, it can be hard for a user to find components when lots of servers are available. Hence, while monitoring systems can offer graphical representation of the physical layout of monitor data (usually hierarchical data), such as illustrated 0, these systems should also provide logical representations, where components are

logically organized (related and grouped) through layers of different services (for a distributed database, for instance, query engines, persistency layer, storage layer, etc.), as illustrated in 0. Logical view are more immune to visual clutter when the system scales up.
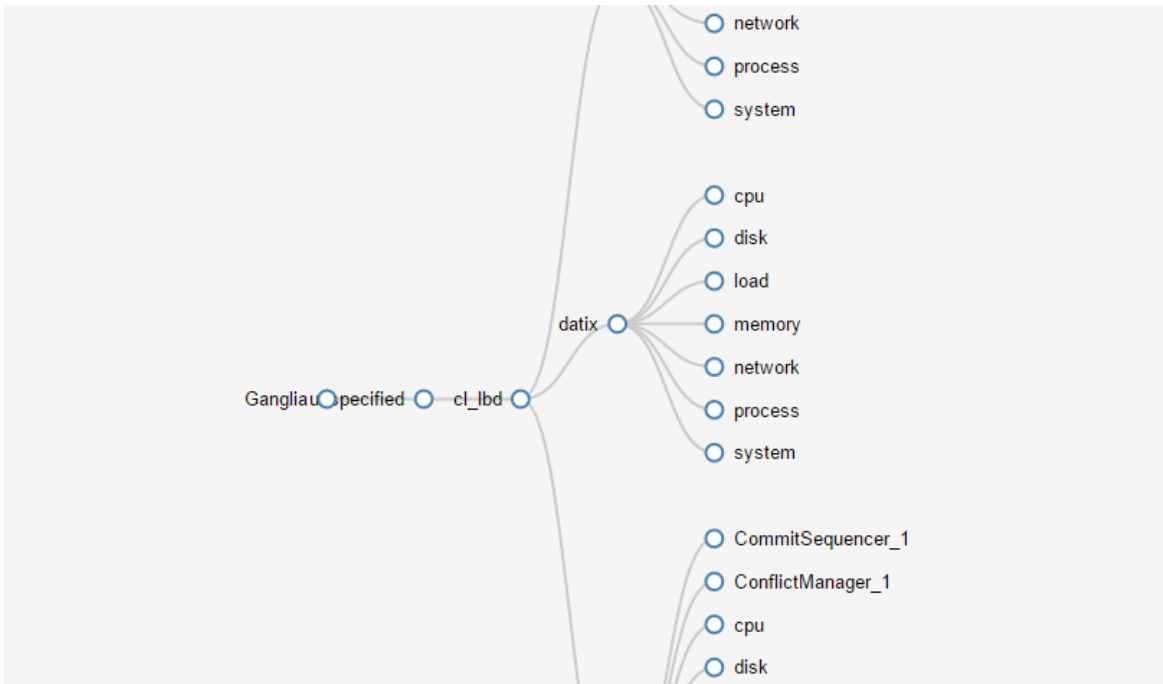

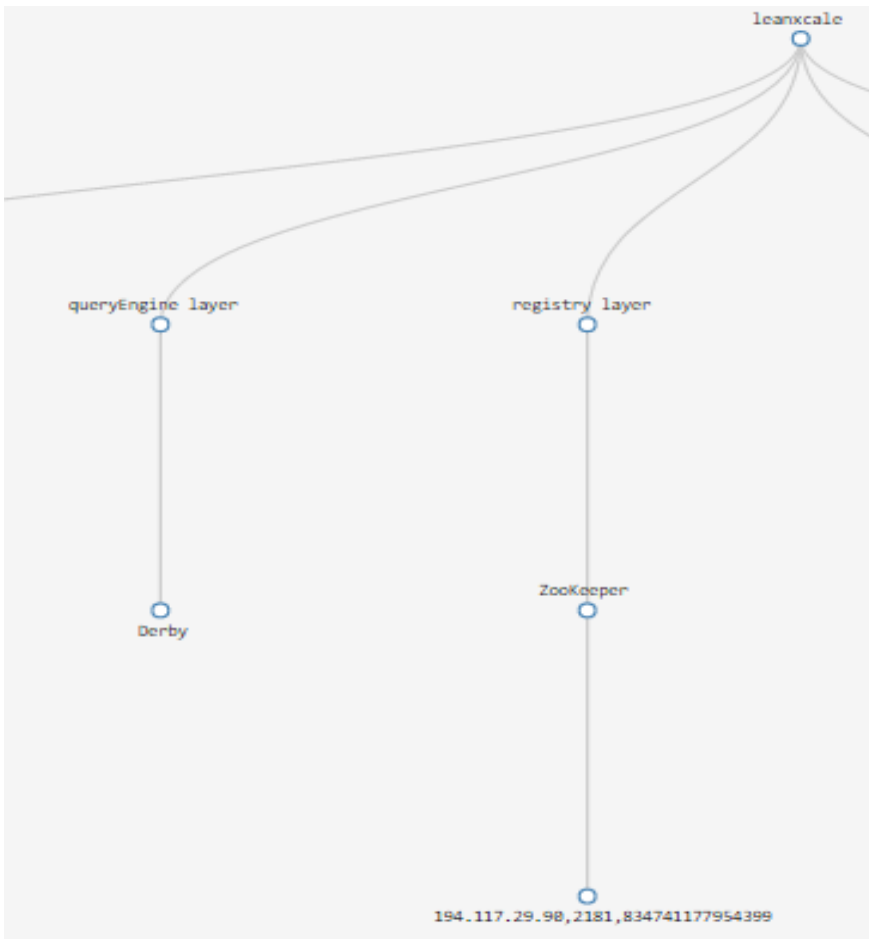**Figure 2.** Hierarchical system view, representing the servers and groups of metrics available

Figure 3. Logical system view, representing services distribution

- Hive plots (Engle and Whalen 2012) are good to display connections between components: the proposed system uses a hive plot to show relations among servers (left axis), metrics (right axis) and groups of metrics (top axis). In 0when the "cpu" group is selected, all the links to machines who analyse that group of metrics and all the links to its metrics are highlighted. It is easy for the user to perceive which groups contain more metrics and which server has more metric inspection.
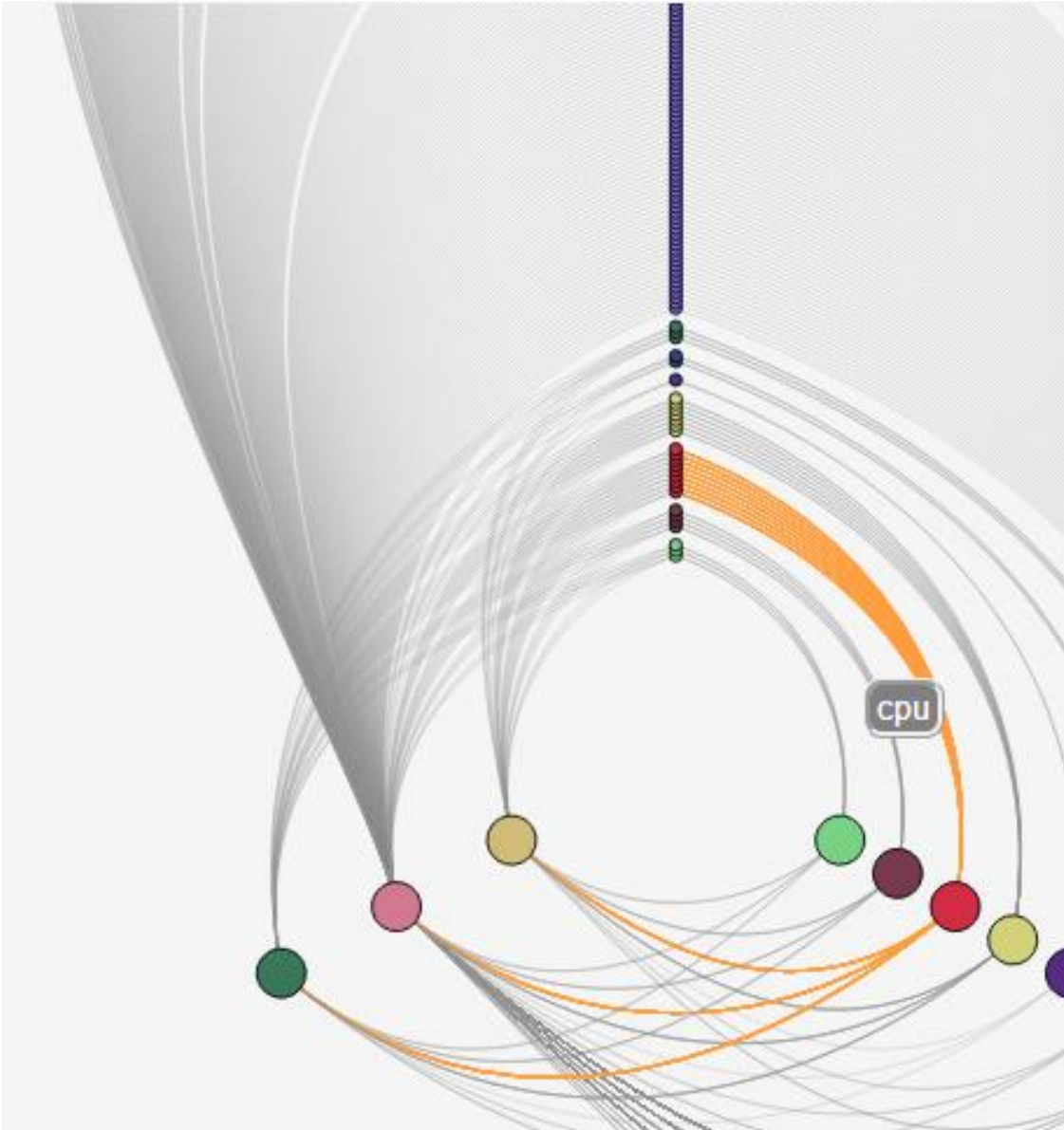

Figure 4. Hiveplot view, representing the connections between servers, groups of metrics and metrics

- Tree maps offer in-depth perception: tree map representation does not offer a representation of the impact of each component in the system but the relevance that it may have according to the number of metrics. This view provides a dimensional perception of each component, which is directly proportional to the number of metrics analysed. An example of this view can be seen in **Hata! Başvuru kaynağı bulunamadı.**
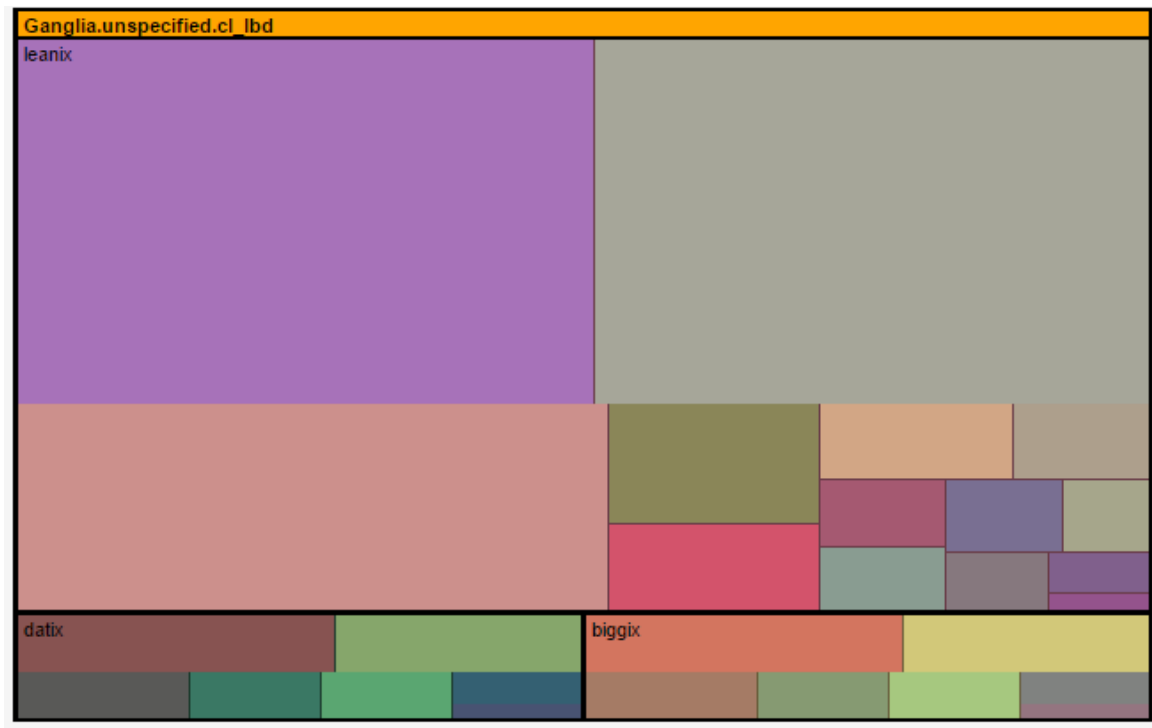
Figure 5. Tree map view, representing the system metrics composition

CONCLUSIONS

Although the market has a considerable number of monitoring tools there are not complete solutions with both backend and frontend features optimized to monitor large clusters. From the review of monitoring tools it is possible to verify they have common characteristics, such as: metrics representation through line charts with the possibility to merge several metrics into a single chart; they provide a reduced set of customizable chart options; visualization of monitor systems is oriented to physical components and not oriented to logical views. On visualization of monitoring data, line charts are good to display linear and time-dependent data. For representation and exploration of the monitored structure there are good alternatives - logical views, tree maps, hive plots - which can represent the system, components and relation among components and help select components which, later, allow the visualization of corresponding metrics.

ACKNOWLEDGMENTS

REFERENCES

A. C. Emanuel Pinho, "Monitoring Systems for Parallel Distributed Data Management Systems," in In Information Systems and Technologies (CISTI), 2016 11th Iberian Conference, 2016.

J. James, "Domosphere," 13 Agosto 2015. [Online]. Available: https://www.domo.com/blog/2015/08/data-never-sleeps-3-0/.

A. McAfee, E. Brynjolfsson, T. H. Davenport, D. Patil and D. Barton, "Big data," The management revolution. Harvard Bus Rev, vol. 90, pp. 61-67, 2012.

M. Zuckerberg, "Facebook," Junho 2015. [Online]. Available: https://www.facebook.com/zuck/posts/10102213601037571. [Accessed 28 Outubro 2015].

M. Chen, S. Mao and Y. Liu, "Big data: A survey," Mobile Networks and Applications, vol. 19, pp. 171-209, 2014.

S. Andreozzi, N. De Bortoli, S. Fantinel, A. Ghiselli, G. L. Rubini, G. Tortone and M. C. Vistoli, "GridICE: a monitoring service for Grid systems," Future Generation Computer Systems, vol. 21, pp. 559-571, 2005.

I. MongoDB, "Monitoring for MongoDB," [Online]. Available: https://docs.mongodb.org/manual/administration/monitoring/. [Accessed 28 Outubro 2015].

T. Xu, "Grid monitoring system survey," 2011.

S. Sreeja and S. Chaudhari, "Study on Grid Resource Monitoring and Prediction," Procedia Computer Science, vol. 45, pp. 815-822, 2015.

S. F. Silva and T. Catarci, "Visualization of linear time-oriented data: a survey," in Web Information Systems Engineering, 2000. Proceedings of the First International Conference on, 2000, pp. 310-319.

M. Massie, B. Li, B. Nicholes and V. Vuksan, Monitoring with Ganglia, O'Reilly Media, Inc., 2012.

M. L. Massie, B. N. Chun and D. E. Culler, "The ganglia distributed monitoring system: design, implementation, and experience," Parallel Computing, vol. 30, pp. 817-840, 2004.

"Observium Docs," Observium, [Online]. Available: http://www.observium.org/docs/. [Accessed 13 Janeiro 2016].

C. Blake, "Observium, the Do-it-All Monitoring Application," Abril 2013. [Online]. Available: http://servernetworktech.com/2013/04/observium-the-do-it-all-monitoring-application/. [Accessed 13 Janeiro 2016].

"Nagios," Nagios Solutions, [Online]. Available: https://www.nagios.org/about/. [Accessed 13 Janeiro 2016].

C. Joseph, "Open Source Server Monitoring Tools – Cacti, Zabbix, Nagios," 8 Agosto 2015. [Online]. Available: https://cloudstats.me/2015/08/08/open-source-server-monitoring-tools-cacti-zabbix-nagios/. [Accessed 13 Janeiro 2016].

E. Simmonds and J. Harrington, "SCF/FEF Evaluation of Nagios and Zabbix Monitoring Systems," 7 Julho 2009.

E. R. Tufte and P. Graves-Morris, The visual display of quantitative information, Graphics press Cheshire, CT, 1983.

C. G. Healey, "Choosing effective colours for data visualization," in Visualization'96. Proceedings., 1996, pp. 263-270.

A. Buja, J. A. McDonald, J. Michalak and W. Stuetzle, "Interactive data visualization using focusing and linking," in Visualization, 1991. Visualization'91, Proceedings., IEEE Conference on, 1991, pp. 156-163.

J. D. Mackinlay, S. K. Card and G. G. Robertson, "Rapid controlled movement through a virtual 3D workspace," in ACM SIGGRAPH Computer Graphics, 1990, pp. 171-176.

N. Iliinsky and J. Steele, Designing Data Visualizations: Representing Informational Relationships, O'Reilly Media, Inc., 2011.

"D3 Data-Driven Documents," [Online]. Available: http://d3js.org/. [Accessed 04 Fevereiro 2016].

S. Engle and S. Whalen, "Visualizing distributed memory computations with hive plots," in Proceedings of the Ninth International Symposium on Visualization for Cyber Security, 2012, pp. 56-63.

"Wikipedia," Wikipedia, Novembro 2015. [Online]. Available: https://en.wikipedia.org/wiki/Wikipedia:Graphs_and_charts. [Accessed 13 Janeiro 2016].

A. Dam, A. S. Forsberg, D. H. Laidlaw, J. J. LaViola Jr and R. M. Simpson, "Immersive VR for scientific visualization: A progress report," Computer Graphics and Applications, IEEE, vol. 20, pp. 26-52, 2000.

Ganglia, "Ganglia Monitoring Systems," Janeiro 2001. [Online]. Available: {http://ganglia.sourceforge.net/. [Accessed 13 Janeiro 2016].