

A Dynamic and Interactive Monitoring System of Data Center Resources

Ling-Fei YU^{1,a}, and Sheng-Er ZHOU²

¹ Hangzhou College of Commerce, Zhejiang GongShang University, Hangzhou, P.R. China

² School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu, P.R. china

^a Corresponding author: linphie@163.com

Abstract. To maximize the utilization and effectiveness of resources, it is very necessary to have a well suited management system for modern data centers. Traditional approaches to resource provisioning and service requests have proven to be ill suited for virtualization and cloud computing. The manual handoffs between technology teams were also highly inefficient and poorly documented. In this paper, a dynamic and interactive monitoring system for data center resources, ResourceView, is presented. By consolidating all data center management functionality into a single interface, ResourceView shares a common view of the timeline metric status, while providing comprehensive, centralized monitoring of data center physical and virtual IT assets including power, cooling, physical space and VMs, so that to improve availability and efficiency. In addition, servers and VMs can be monitored from several viewpoints such as clusters, racks and projects, which is very convenient for users.

1 Introduction

Cloud Computing[1] can be classified as a new paradigm for the dynamic provisioning of computing services supported by state-of-the-art data centers that usually employ Virtual Machine (VM) technologies for consolidation and environment isolation purposes [2]. A data center can be seen as a virtual supercomputer which integrates large sets of computer systems and associated components, such as telecommunications and storage systems, power supplies, environmental controls (e.g., air conditioning, fire suppression) and security devices. Increasing efficiencies, reducing costs and maintaining availability are top priorities for today's data center administrators. To maximize the utilization and effectiveness of resources, it is very necessary to have a well suited management system for modern data centers. Traditional approaches [3-5] to resource provisioning and service requests have proven to be ill suited for virtualization and cloud computing. The manual handoffs between technology teams were also highly inefficient and poorly documented.

The management system should centralize real-time monitoring and help for intelligent capacity planning, high-fidelity visualization of a data center's critical systems. It can help data center managers identify and eliminate sources of risk to increase availability of critical IT systems. A lot of the discussion in this area has focused on energy issues, but other metrics beyond the Power usage effectiveness (PUE) can give a more detailed picture of the data center operations[6]. These intrinsic risks have resulted in an increase in market demand for integrated monitoring and management solutions capable

of "bridging the gap between IT and facilities" systems[7]. It leads to the development of "intelligent capacity planning" solutions that support synchronized monitoring and management of both physical and virtual infrastructures. More effective automation tools can also improve the number of servers or virtual machines that a single admin can handle.

Existing solutions in data center operating, however, has several problems. Firstly, while technologies such as cloud computing and software defined networking (SDN) [8] provide methods for dynamic management, data center runtime status data, on which dynamic management is based, usually received little attention. Secondly, a technology only focuses on a single domain, e.g., VM management in cloud computing and network management in SDN may not be able to offer an optimal solution. We argue that information collected in each domain should be shared and control methods enabled by individual technologies should be combined. And thirdly, existing solution such as Ganglia [9] is widely used, but it is mainly focused on host servers organized in clusters and grids. No runtime information about VMs, power supplies and cooling system is collected and managed. And its data storage and visualization is using RRD tool, which is difficult for function extension and user customization[9]. Another management system CloudStack mainly manages the network, storage, and computing nodes, and is more focused on configuration of a data center. Runtime metric status monitoring is not covered [10].

In this paper, we present ResourceView, a dynamic and interactive monitoring system for data center resources. By consolidating all data center management

functionality into a single interface, we delivers the complete data center monitoring system for the 24/7 data center, shares a common view of the timeline metric status, provides comprehensive, centralized monitoring of data center physical and virtual IT assets including power, cooling, physical space and VMs to improve availability and efficiency, proactively monitors and maintains server health to support faster troubleshooting. So users can simply plan and manage your data center in a more intelligent way. In addition, servers and VMs can be monitored from several viewpoints such as clusters, racks and projects. It's very convenient for users. With animated transitions and well-crafted interfaces, ResourceView makes exploring data feel more like playing a game. Interactivity can also encourage engagement with the data in ways that static images cannot. It can help users see the world in a new way, revealing unexpected patterns and trends, making intelligent decisions.

The remainder of the paper is organized as follows. In section 2, we introduce background. Our system design and implementation are presented in section 3 and section 4, respectively. At last, we conclude our work in section 5.

2 Background

ResourceView is our intelligent capacity planning solution that deals with the monitoring, collection and analysis of real-time, continuous and accurate information from an intelligent data center infrastructure that allows for immediate action. It's designed to keep the data center in an optimal state of balance between availability, efficiency and capacity.

The basic of the whole solution is to gain and present an integrated real-time holistic view of the entire data center infrastructure. Real time data presented in an integrated view allows you to diagnose individual components without having to walk the floor. It can make intelligent assessments of your assets so you can address problems and proactively prevent issues happening. Consequently, you can make faster and smarter decisions that will align data center performance with your business goals.

ResourceView is a data center management system to collect sufficient information, providing user-friendly interfaces for visualization and control of the whole system, and provide an extensible platform to implement various system tuning strategies so as to better utilize a data center infrastructure.

We will provide five functions to build a generic data center management system:

1) Data Collection: This function focuses on collecting data center equipment status for monitoring, analysis, visualization, and management;

2) Data Storage: This function focuses on storing data efficiently and providing effective API for data visualization and data mining;

3) Data Mining: This function focuses on analyzing collected data and provide solutions to various applications, for example, green data center.

4) System Monitoring: This function visualizes the data in web pages and a 3D animation system.

5) System Management: This function mainly focuses on managing physical servers, VMs, applications and network devices, such as VM immigration.

A data center infrastructure can be roughly divided into several components, including PDUs for power supply, air conditioners for cooling, servers and networking devices for ICT service provisioning. Challenges arise to manage and optimize these devices as a whole system, since these devices provide inherently different methods for monitoring and control. Most devices provide command line interfaces (CLIs) and web user interfaces (WUIs) via computer network. To shield the differences among these various devices, Data Collection is responsible to provide a unified API to collect all the information of various devices and store into the database. Meanwhile, we must provide a unified comprehensive user interface for users to monitor and manage all the data center devices. As a solution for web based management, we present ResourceView system in this paper. Data Collection design is beyond the scope of the paper.

3 Design of ResourceView

3.1 High Level Design

To develop the dynamic, interactive data-driven web application ResourceView, we chose the popular stack which includes the Linux operating system, Apache web server, and scripting language PHP. There are so many candidates for database, like MySQL, MongoDB[11] and many others. Considering its maturity and open-source characteristics, MongoDB is our current choice of Data Collection Subsystem. As growth of data volume and rising of other constraints, for example, performance, the choice may evolve. Hence, database access interfaces should be designed to shield differences between different database options and provide a unified access interface ideally (or work with little code changes). Since data collecting and storing into the database is beyond the scope of this paper, ResourceView can be divided into the following layers.

3.1.1 Database Access Layer

Database access layer encapsulates all the related interfaces for accessing database, such as acquiring the cloud hierarchy structure, acquiring the latest metric information of resources utilization and host server configuration information. PHP MongoDB Driver is used here to provide low-level interfaces for accessing our MongoDB database from the PHP code on the server side. Users can request to view the metric data of any time range freely, even the whole year. But in practice, we must sample the mass data. Otherwise, it may results in memory overflow or network congestion, since a huge amount of data is transferred between web server and database server.

3.1.2 Data Communication Layer

Data Communication Layer is responsible to handle the data communication between the PHP pages on the server side and the JavaScript code on the client browser side. After getting data from the database on the server side, the data of information should be sent to the client to feed the graph engine and then the data can be displayed. Especially, the web application should be able to make quick, incremental updates to the related charts without reloading the entire browser page. This makes the application faster and more responsive to user actions.

The implementation of this layer will include two parts: the server side and the client side. In our design, JSON is the exchanging data format between the server and the client. When the web page loads, or the charts need automatically refresh, the client will start an Ajax request for the data. As response, the server will then acquire the related data from database, encode it into JSON format, and send back as the Ajax response to the client.

3.1.3 Data Pre-processing Layer

The data from Ajax response actually may not directly be applied for the graph engine. So there is Data Pre-processing layer, to handle the data received from Ajax response. The functions may include data transforming, setting the label and color of each kind of data, setting axis formats and things like that. All these data pre-processing things usually depend on the data visualization techniques, or say the graph engine. So there comes Data Presentation Layer.

3.1.4 Data Presentation Layer

Data Presentation layer is responsible for drawing data-driven attractive and perspicuous charts. The key of the layer is the graph engine. There are bundles of choices for this. Here we choose D3[12] and HighCharts[13] because of their good performance and attractive user experience.

3.2 Metric Data Parameters

The goal of our web system is to eliminate isolated island of information, to make data center administrators easily and efficiently get desired information. In the integrated user interface, administrators can get one or more types

of data including CPU workload, CPU utilization, memory utilization, network utilization for physical servers and VMs, power and environment information. Parameter means our collected data from data center. Since we have collected a lot of data, and many of them may not be used in practice. We only display parts of them which users are most interested in. All of this information is modelled in a single data structure with an extremely high level of fidelity so any combination of devices (or even the data center as a whole) can be studied and actively managed using various planning scenarios.

4 System Implementation

We are running several data centers on a scale from dozens of servers to hundreds of servers. Now we have deployed this dynamic monitoring web system ResourceView for the data center infrastructures management.

4.1. Main User Interfaces

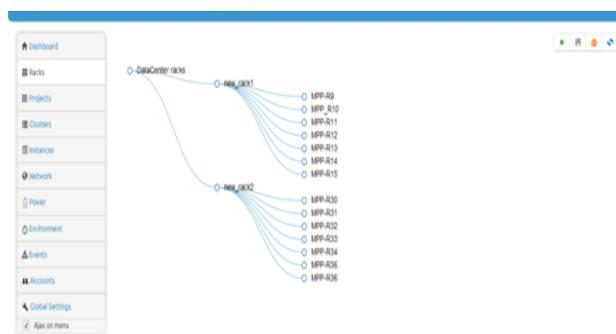


Figure 1. Tree View of Data Center

We provide three different views on our web pages: tree views, static views and dynamic views. In a data center, large sets of diverse geographically distributed servers can be seen from different viewpoints: Rackspace, Grid, and Project. So respectively, servers can be grouped into different racks, or into various grids, or in various projects. Our web system delivers the administrators all the three viewpoints to view the entire data center, and to locate any one of the servers and VMs. The tree view is used to see each hierarchy structures of a data center organized in the three viewpoints.

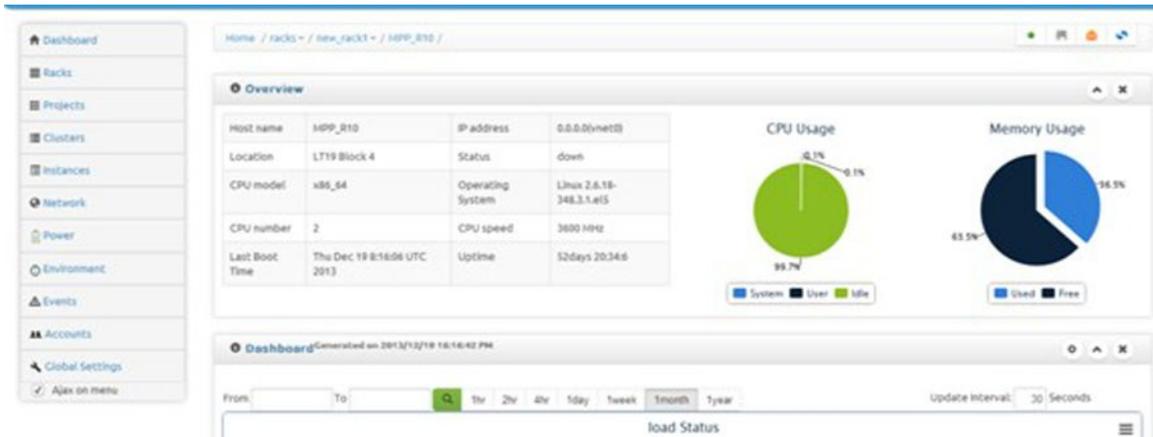


Figure 2. The Static View of A Node

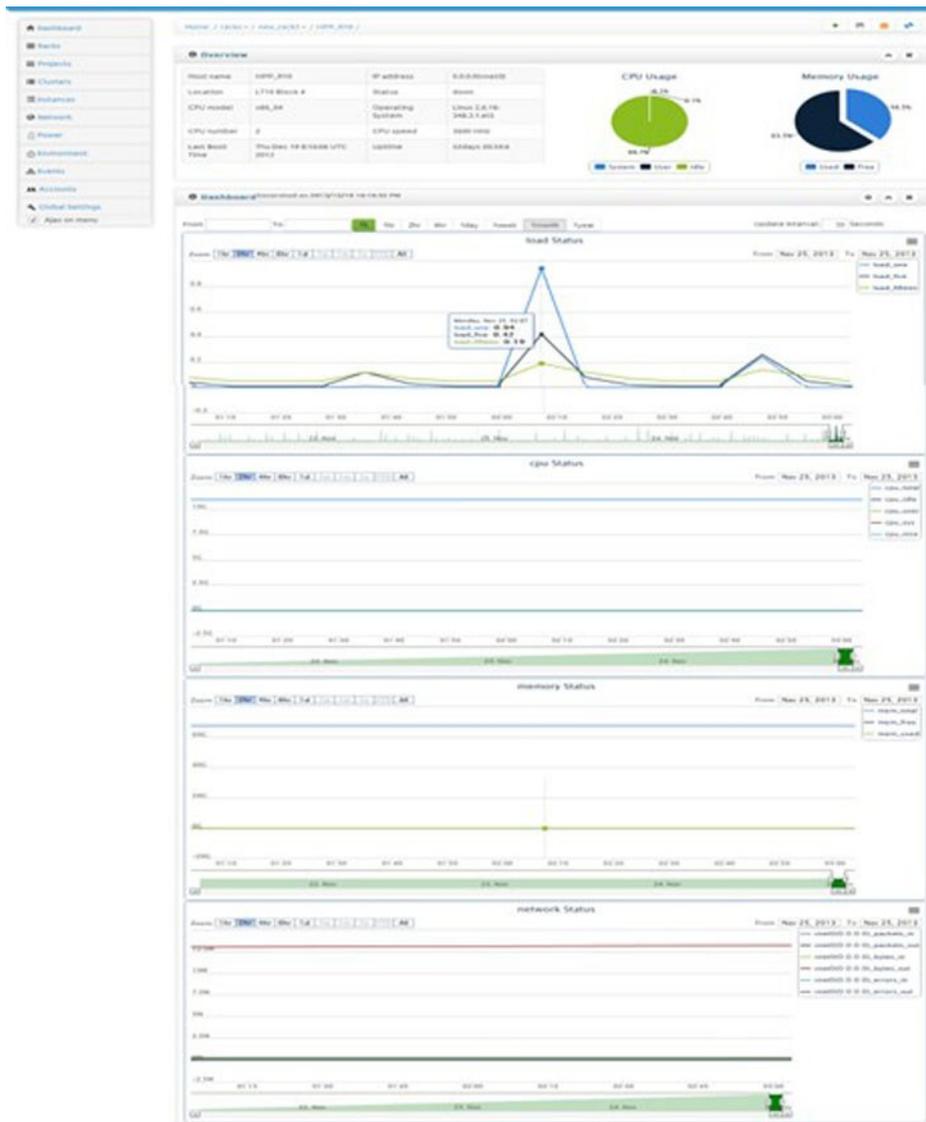


Figure 3. The Dynamic View of A Node

It's very convenient and important for users to get a clear whole image of a data center. Users can quickly find out the location relationship and relocate to the integrated monitoring page of a specific node. The tree is displayed in a dynamic collapsible layout. On the tree, users can click on a node dot to expand or collapse a subtree. More

importantly, users can click on a node name to access the node's integrated monitoring page. For example, a tree of the data center racks is shown in Fig. 1.

Fig. 2 and Fig. 3 are the static view and dynamic view of a specific node. They provide homogeneous views of the underlying heterogeneous data. The static view

simply displays the configuration or overall information of a server or VM in a well-organized table, including name, IP, location, status, CPU model, CPU number, CPU speed, operating system, last boot time, uptime, and things like that, as shown in Fig. 2.

The dynamic view shows pie charts of the latest CPU and memory usage, and also the animated charts of timeline metric parameters like CPU average workload, CPU usage, memory usage, network utilization. The charts can automatically refresh according to the update interval set by users. The time cycle can be different from the data collecting cycle. From the single page, users can easily know all about the specified node, and help users make intelligent decisions.

4.2 Main Features

The main features of ResourceView are as follows:

4.2.1 Quick search any one of the data center resources

A data center usually includes dozens, hundreds, or even thousands of servers. In this project, we provide several quick ways to find any one of the data center nodes and relocate to the node's monitoring page. One fast way is to use the search function in the top bar. Also, you can use three kinds of data center hierarchy trees to find a node.

4.2.2 Different query time range

We provide a group of quick query ways for viewing metric data, including the last 1 hour, 2 hours, 4 hours, 1 day, 1 week, 1 month, 1 year and user customized query time range;

4.2.3 Auto Refresh time-series data charts

The metric status charts can automatically refresh according to the update interval set by users. The time cycle can be different from the data collecting cycle.

4.2.4 Enable/disable data series line in a chart

Users can directly disable or enable the display of a data series line in a chart by clicking on the label or line. It's very convenient for users to just compare part of the data parameters.

4.2.5 Tracking tooltip

On hovering a chart, it can display a tooltip text with information on each point and data series. The tooltip follows as the user moves the mouse over the graph,

making it easy and intuitive to read out the values from the chart.

4.2.6 Navigating Range selector

While navigating a large dataset, perhaps hourly values over several years, it is important for the charting library to provide a quick way to view a certain time range. Highstock provides a range selector where the user can zoom in on preselected time ranges like 1m, 1y or YTD, or manually add date ranges.

4.2.7 Multiple Axes

Sometimes there may be several data series that are not the same scale, such as packets_in, packets_out, bytes_in, bytes_out may be in different units but all of them are put in a network utilization chart. We may provide respective y axis for each series.

5 Conclusions

In this paper, we present an interactive dynamic monitoring system of a data center; provide users more friendly and perspicuous ways to monitor the runtime status of various resources in a data center. The paper describes the system architecture, design and implementation. We are planning to improve its functions and make a more intelligent system in the future, such as adding the trigger mechanism to connect user actions on the page and the low-level action engine system.

Acknowledgement

This work is supported in part by Zhejiang Provincial Natural Science Foundation of China under Grant No. LQ16F02001.

References

Here are some examples:

1. R Buyya, CS Yeo, S Venugopal, Cloud computing and emerging IT platforms: Vision, hype, and reality for delivering computing, Future Generation Computer Systems, 25(2009)
2. B Barham, S Fraser, A Harris, A. Warfield. Xen and the Art of Virtualization, Proc. Acm Symposium on Operating Systems Principles, 36 (2003).
3. K Chen, W Sun, L Zhu, et al. An Adaptive Page-Replacement Strategy for Spatial Database Systems. Journal of Computer Research and Development 10 (2011): 018
4. J Bi, H Yuan, W Tan, Application-Aware Dynamic Fine-Grained Resource Provisioning in a Virtualized Cloud Data Center, J. IEEE Transactions on Automation Science and Engineering, (2015).
5. W Song, Z Xiao, Q Chen, H Luo, Adaptive Resource Provisioning for the Cloud Using Online Bin Packing, 63 (2013)

-
6. DL Beaty, DL Beaty, Data center energy metric: Power Usage Effectiveness (PUE), J. Ashrae, 2013.
 7. Data center infrastructure management , on http://en.wikipedia.org/wiki/Data_center_infrastructure_management
 8. N Feamster, J Rexford, E Zegura, The road to SDN: an intellectual history of programmable networks, J. Acm Sigcomm Computer Communication Review, 44(2014)
 9. Ganglia, on <http://ganglia.sourceforge.net/>
 10. CloudStack, on <http://cloudstack.apache.org/>
 11. MongoDB, on <http://www.mongodb.org/>
 12. D3.js, on <http://d3js.org/>
 13. Highcharts, on <http://www.highcharts.com/>