An Architecture to Evaluate Scalability, Adaptability and Accuracy in Cloud Monitoring Systems

Guilherme da Cunha Rodrigues, Gléderson Lessa dos Santos, Vinícius Tavares Guimarães,
Lisandro Zambenedetti Granville, Liane Margarida Rockenbach Tarouco
Computer Networks Group
Institute of Informatics
Federal University of Rio Grande do Sul
Porto Alegre, Brazil
Email: {gcrodrigues,glsantos,vtguimaraes,granville,tarouco}@inf.ufrgs.br

***Abstract*— In a cloud computing environment, multiple resources need to be properly managed to improve resource utilization and offer predictable performance to customers. One essential management function that gains a special importance in the context of clouds is monitoring. Nowadays, cloud monitoring solutions need to meet several requirements such as Scalability, Accuracy, and Adaptability. In This paper, we proposed an architecture to evaluate Scalability, Accuracy, and Adaptability in cloud monitoring systems based on local filters. The results show a mutual influence among these requirements. Moreover, the proposed architecture is simplified and the filtering methods show promising results to Scalability and Adaptability.**

I. INTRODUCTION

Cloud computing is a consolidated technology to provide computing resources on demand to end-users [1]. In order to support different customers and usage profiles, a cloud provider offers a variety of resource types (*e.g.,* application, storage, network) and sizes. In a cloud computing environment, multiple resources need to be properly managed to improve resource utilization and offer predictable performance to customers [2]. One essential management function that gains a special importance in the context of clouds is monitoring [3].

Nowadays, cloud monitoring solutions [4] [5] support many functionalities (*e.g.,* analyzes, notifications, SLA control) and can be used to monitor resource and service in a cloud. However, in order to adapt to the dynamics of a cloud and work properly, monitoring solutions need to meet some requirements such as scalability, elasticity, comprehensiveness, accuracy, and adaptability [6] [7].

Scalability is a key requirement in cloud computing to support an increasing number of customers and applications [8] [9]. Thus, the cloud monitoring solution must also be scalable in several ways to work properly with a large number of resources, large number of users (*e.g.,* service providers, customers), and with different types of information (*e.g.,* physical data, virtual data). In this way, the cloud monitoring solution needs to be scalable also to ensure that it can cope with a large number of probes and is able to collect, transfer, and analyze huge volumes of generated data [7] [10].

Currently, some important research efforts aim to address scalability in cloud monitoring solutions [10] [11] [12]. Usually, these works have addressed scalability in terms of the number of resources and end-users that can be monitored at the cost of an increase in the monitoring traffic. Such growth in the monitoring data can create difficulties to network administrators that need to analyze such data. To tackle such problem, cloud monitoring solutions usually employ methods such as aggregation and filtering for monitoring data [12].

In general, aggregation and filtering are designed for monitoring complex architectures and use different methods to compress and reduce the monitored data. In filtering, for example, monitoring solutions usually apply filters by slice, resource type, or statistics. Filtering can thus decrease the impact of monitoring data transfer over the network load. However, filtering reduces the accuracy of the monitoring process. Accuracy is affected by three main reasons. First, there is less information to be analyzed. Second, the complexity of the monitoring architecture slows down the analysis process. Finally, in current monitoring architectures, traffic is replicated among different layers, which can also overload the network.

In this context, there is a clear trade-off between accuracy and scalability. In addition, scalability is closely related with another requirement, adaptability. This occurs because if scalability is fulfilled, then a cloud can add resources on demand; the cloud monitoring system then needs to add probes to attend this demand. For this reason, the number of agents/probes increases, thus generating more monitoring data and influencing over adaptability. Therefore, there is also a clear trade-off between adaptability and scalability.

In this scenario, there are several investigations assessing and evaluating cloud monitoring requirements. However, these investigations have not considered the influence of a specific requirement over another, and vice versa. Therefore, there is a lack of knowledge about the mutual influence among scalability, accuracy, and adaptability. In this paper we propose a simplified, scalable, experimental, filtering-based cloud monitoring architecture, for application in large-scale cloud environments. Our generic architecture reduces communication among layers, minimizes network overload, and improves the scalability. We utilized our experimental architecture in order to analyze the mutual influence among scalability, accuracy, and adaptability.

The remainder of this paper is organized as follows. In Section II, we present related work. In Section III, we show the architecture proposed to improve scalability in cloud monitoring solutions and to analyze the mutual influence among scalability, accuracy, and adaptability. In Section IV, we present our evaluation, while in Section V we finally conclude this paper with conclusions and future work.

II. RELATED WORK

In this section, we presented some works that somehow have influence of scalability, accuracy, and adaptability.

Scalability is one of the main requirements for cloud monitoring solutions. Nowadays, a large number of cloud monitoring solutions have concerns about scalability issues [11] [12]. As we mentioned in the previous section, to reach scalability researchers/developers count with complex architectures with techniques known as aggregation and filtering [7]. A functional architecture is proposed by Hasselmeyer and Heureuse [11]. The functional architecture has a generic event notification propagation system called Message Bus that receives monitoring information, on demand or periodically, from specific agents located around the infrastructure. The solution proposed has an engine to aggregation and filtering that operates in a layer over the Message Bus. This proposed is functional but it generates replicated monitoring traffic between agents/message bus and also over the message bus/engine. The Information Management Overlay (IMO) presented by Clayman *et al.* [12] is a fundamental component of an autonomic network architecture show by Galis *et al.* [13]. IMO is a management infrastructure represented by a controller that interacts with virtual networks and collects information from/to virtual network devices. To improve the collection and transferring of information, IMO controller utilizes a solution based on IMO-Nodes that use techniques of aggregation and filtering.

Accuracy is an important cloud monitoring requirement closely related to issues such as performance, costs, and SLA. Currently, there are studies related to accuracy, associating this requirement with different metrics such as CPU utilization [14] [15], and cloud services [16]. However, these works present results that need to be accurate, but, none of these works have concerns with the effective accuracy of used monitoring solutions, as well as the influence of other requirements over accuracy. An evaluation closely related to accuracy is presented by Osterman *et al* [14]. Osterman *et al.* uses the Amazon EC 2 [17] to provide a study to analyze the utilization of clouds in order to reduce operational costs to researches that needs high performance computing. This work show a comprehensive evaluation about basic metrics (*e.g.,* CPU utilization, memory). Although, Osterman *et al* presents an evaluation that needs to be accurate, because the results can inferred in reduced costs. The accuracy issues are not mentioned. Another example is presented by Hill and Humphrey [15] that uses amazon EC 2 also as a cloud environment scenario to basic metrics evaluation in the context of high performance computing. Miam *et al* [16] have as objective determines the most cost-effective configuration for a given data analytic workload in a cloud. However, Miam *et al* found out as a challenge the develop accurate performance prediction models using standard methods. Thus, accuracy emerges as a secondary problem that needs to be solved in this work.

Nowadays, adaptability emerges as a key cloud monitoring requirement. Nevertheless, the evaluation of adaptability is not trivial on monitoring systems, as well as its direct correlation with scalability and accuracy. However, there are some papers that provides evidences related to adaptability [18] [19]. Monalytics [18] allows to configure its agents according to the features of a monitoring system in real time. In this way, a monitoring system could be adjusted in order not to be invasive. Wang *et al* [19] shows a flexible architecture that operates aiming to reach a cost effective and performance solution to manage a large-scale environment such as a cloud. Therefore, using the same approach of Monalytics, Wang *et al* wants to implement an environment where a monitoring system is adjusted in order not to be invasive.

Finally, we proposed an architecture to address some problems presented in this section (*e.g.,* replicated monitoring data, complex architecture). Moreover, we proposed an approach that wants to evaluate the requirement and its influence over other requirements.

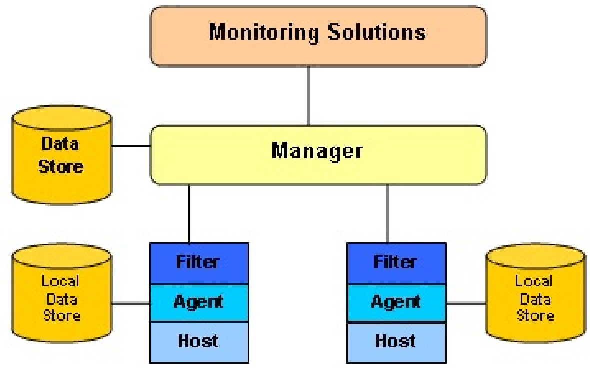

Figure 1. Proposed Architecture.

III. Architecture

An experimental architecture is proposed to improve scalability and it is used to evaluate the mutual influence among scalability, accuracy, and adaptability. The proposed architecture is presented in Figure 1.

The experimental architecture presented is composed of agents, local data base and local filters (called filter in Figure 1) allocated in hosts, where a host represents any type of resource (*e.g.,* virtual machine, application). In a layer above is presented a manager, this layer can update data store and it can communicate with monitoring solutions (*e.g.,* Nagios [20], Cacti [21]). To properly discuss about the proposed architecture and its features we divided the architecture according to its components as follows in the next subsections.

A. Host, Agent and Local Filters

The host is a monitored entity and it represents cloud computing resources that are monitored by agents. It is worth mentioning, in a host can be allocated agents as much as necessary to monitor all resources.

In turn, agents are responsible for gathering all kinds of data in a host. Closely related to agents are local filters. The local filters have performed activities such as filtering, and on-demand sends monitoring data (samples) to the manager.

The filtering is a method to refine information sent out by agents to the manager. In the architecture, we propose a kind of filtering that operates periodically with samples gathering of agents. These samples are stored in a local database and send on-demand to the manager. The filter works selecting data between low threshold and high threshold. Such thresholds are defined according to percentage of monitoring data refined. This percentage can be adjusted aiming to reduce the network load caused by monitoring data, relieving the network traffic. The filtering could vary from mild to stringent, where the percent of refined data are exemplified through the most extreme examples of mild and stringent.

If the filter is extremely mild, the low threshold is lower and the high threshold is higher. The Figure 2 shows a filter extremely mild that operates discarding 20% of monitoring data and using only the monitoring data between 10% and 90% discarding both the 10% lower values and 10% higher

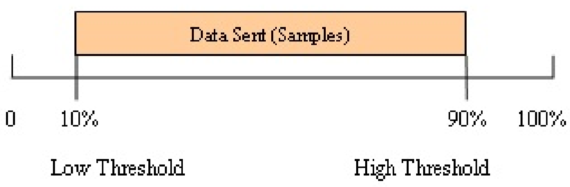

Figure 2. Filtering extremely mild.

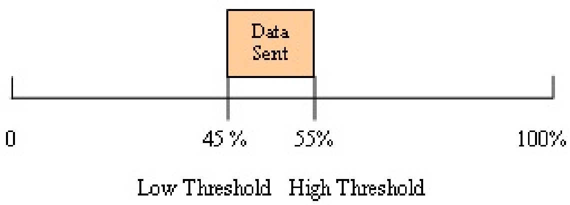

Figure 3. Filtering extremely stringent.

values of monitoring data. If a filter is extremely mild, it is adjusted in order to influence softly over the normal network traffic.

On the other hand, if the filter is extremely stringent, thresholds are more restricted, thereby, the low threshold is higher and the high threshold is lower. The Figure 3 shows a extremely stringent filter that operates discarding 90% of monitoring data and uses only the monitoring data between 45% and 55% discarding both the 45% lower values and 45% higher values of monitoring data. If a filter is extremely stringent, it is adjusted in order to give more advantages to reduce network traffic influencing severely over the amount of monitoring data transferred.

To explain an operation with filters, we utilize an example of measurements to processing. Considering measurements of CPU usage in a virtual machine have provided 60 samples in 1 minute using an extremely mild filter. The filter has select 48 samples between low threshold and high threshold discarding the remaining samples from a local data base.

In this context, intuitively, the filtering assists to reduce the amount of management information through networks, allowing for a monitoring system to operate with a huge number of hosts and agents. For this reason, the filtering could be used to reduce the impact of the scalability over the network overload. However, filtering could impair accuracy because the monitoring system operates with less information. In section IV, we present an evaluation to analyze the scalability and its influences over accuracy and adaptability.

B. Manager and Aggregation Device

The Manager receives monitoring data from filters and organizes the data on a global data store, interacts with monitoring solutions, and organizes monitoring data according to type using aggregation. First of all, a manager organizes the data on a global data store aiming to centralize the monitoring data gathered by agents. Thus, monitoring solutions can make requisitions to managers in order to obtain monitoring data.

In addition, a manager organizes monitoring data using aggregation according to type. The aggregation is a method to join the same type of data into the same group of information. In the architecture, the main function of aggregation is organizing data of the same type originating from several agents spread in a cloud computing environment. Thus, this data can be analyzed jointly allowing to the manager to have enough information to observe the global state of the cloud environment. For instance, considering several measurements of CPU usage from several hosts organized into one group of data showing the global CPU utilization to a host group.

C. Monitoring Solutions

Monitoring solutions are independent solutions of the proposed architecture. These solutions could use the proposed architecture to improve its operation in issues such as scalability, and adaptability. Monitoring solutions are tools used to monitor traditional systems or cloud computing environments such as Nagios [20]. In the architecture, monitoring solutions obtains refined monitoring data from manager and global data store.

In general, monitoring solutions are used by cloud operators (*e.g.,* customers with different profiles, service providers, network administrators). Therefore, different types of cloud operators might utilize monitoring solutions that provides its specific necessities. In this way, different types of monitoring solutions can be used to provide several types of monitoring services (*e.g.,* monitoring solutions to SLA, monitoring solutions to application).

IV. Evaluation

The evaluation is intended solely for analysis to each one of the requirements in an isolated way over the experimental architecture. For instance, to accuracy evaluation are used data previously collected that were stored. This method could affect other requirements directly such as timeliness [19]. However, to accuracy is a relevant form for evaluation because allows to analyze the same group of data in several ways (*e.g.,* without filter, by filters with different thresholds, by different types of filters).

In order to evaluate the proposed architecture for cloud monitoring requirements such as accuracy, scalability,and adaptability. We divided the evaluation according to specific requirement as follows in the next subsections.

A. Accuracy

In accuracy evaluation, we use measurements of CPU utilization as an example. NAS Parallel Benchmarks (NPB) [22] are used to create processing load in seven Xen virtual machines [23] (hosts) and two sections of measurements are made. In the first section, we defined a low processing load. In the second section, we defined more processing load aiming to provide broader measurements. To each virtual machine are gathered 300 samples of CPU utilization taken at 5 minute intervals and without a filter.

The presented filtering methods are applied over gathered data and, finally the average of CPU utilization is calculated for all gathered data without filter and with several different thresholds organized by low threshold - high threshold as follows, 10% - 90%, 15% - 85%, 20% - 80%, 25% - 75%, 30% - 70%, 35% - 65%, 40% - 60%, 45% - 55%. In this scenario, average of CPU utilization can be used to evaluate how close are the averages for without filter (correct accuracy data) and

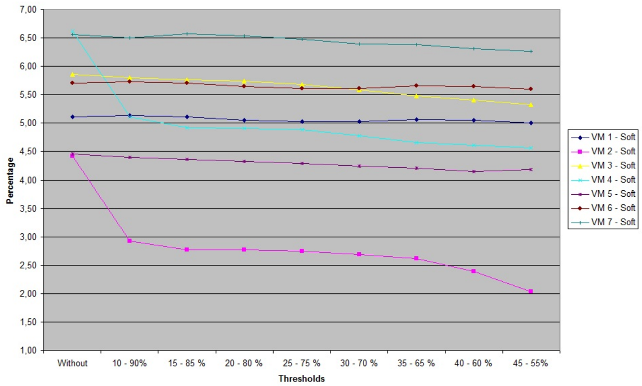

Figure 4. Accuracy Evaluation.

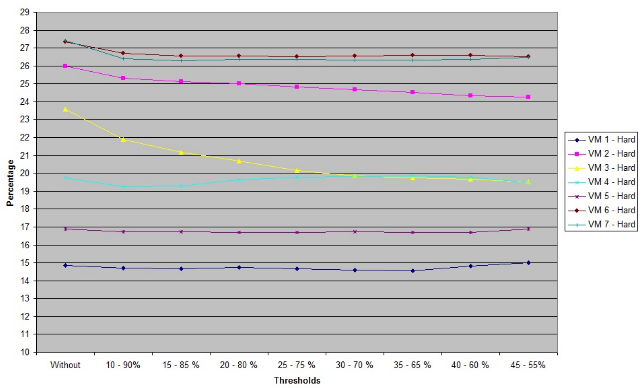

Figure 5. Accuracy Evaluation II.

to all different thresholds. In this way, evaluating accuracy to several sizes of filters.

The Figure 4 shows results of low processing load in the first section. The results show that values to VM 1, VM 3, VM 5, VM 6 and VM7 are closed, but still inaccurate. In general, the most cases show values of percentage to CPU utilization smaller that without filter and present a trend to fall according to the degree of restriction of the filter. In turn, to VM 2, and VM 4 the values are different with more inaccuracy. However, the trend to fall according to the degree of restriction of the filter are maintained.

On the other hand, in Figure 5 are presented the average of CPU utilization in the second section with intensive use of CPU. The results show that average value of monitoring data where was applied any type of filter tends to fall than an average value of monitoring data without a filter. In addition, the VM 3 shows a situation where it is more accented and the gap among the average value of monitoring data is bigger than other VMs. Therefore, it is notorious, the accuracy for the monitoring data average is negatively affected in both test sections.

Additionally, Figure 6 presents all samples to CPU utilization to VM 3 in order to demonstrate that a behavior of the CPU resource has some usage peaks where the application of the proposed filters can usually provide a loss of important values for accuracy requirement, such as peak usage.

The results may seem discouraging, meanwhile, the values usually are closed off right value. For this reason, this type of

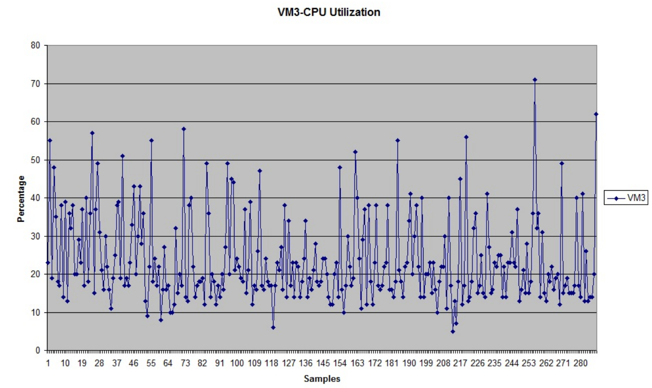

Figure 6. All samples to VM 3

filter could be a good choice for cloud monitoring solutions that do not focus on the accuracy requirement as the main goal. However, the most important issue in this regard is that if any type of filter is applied over a set of monitoring data, the accuracy requirement tends to be impaired.

B. Scalability

In scalability evaluation, we analyze the amount of monitoring data transferred among seven hosts and a manager. In this context, we selected all packages of Simple Network Management Protocol (SNMP) [24] aiming to calculate the average of monitoring data in the network. Thus, we defined the value of 62 bytes as average to monitoring data in this network and this average will work as a reference to analyzes presented in this section. The average value is applied to calculate the amount of monitoring data transferred on a network composed of several hosts. The calculations were made for 1000, 2000, 3000, 4000, and 5000 nodes.

The presented filtering methods are applied without filter and with several different thresholds organized by low threshold - high threshold as follows, 10% - 90%, 15% - 85%, 20% - 80%, 25% - 75%, 30% - 70%, 35% - 65%, 40% - 60%, 45% - 55%. These filters are used in order to show a growing influence of filtering when a cloud monitoring system scales in amount of nodes and agents/probes. In this scenario, filtering operates to reduce the amount of monitoring data in the network. In this way, assisting to decrease the impact of monitoring data over the network performance.

The Figure 7 shows results to 5000 nodes that generates more than 300000 bytes of monitoring data (*i.e.,* see reference without filter). In addition, we can observe the growing influence of filtering over monitoring data. For instance, to threshold 10% - 90% the monitoring data are less than 260000 bytes, to threshold 25% - 75% the monitoring data are less than 160000 bytes, and to the most stringent case, threshold 45% - 55%, the monitoring data are less than 40000 bytes.

Additionally, the Figure 8 presents a behavior analyzis about the influence of filtering over scalability where we can observe that as the nodes increase, monitoring data also increase, and consequently its influence on the network grows. Moreover, the Figure 8 shows the filters assisting to decrease the influence over the network load through reducing the amount of monitoring data as we can observe in the analyzes to each type of filter (*i.e.,* 10% - 90%, 25% - 75%, 45% - 55%).

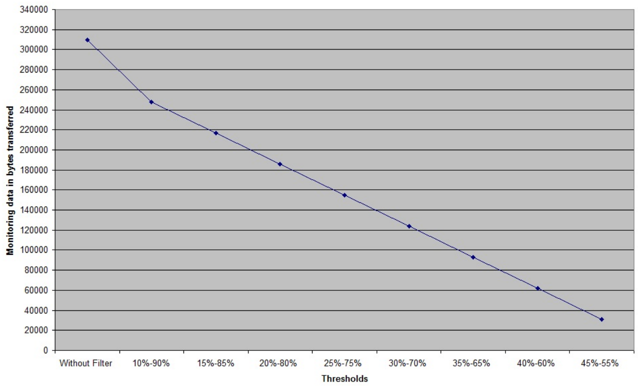

Figure 7. Evaluation to Scalability and Adaptability

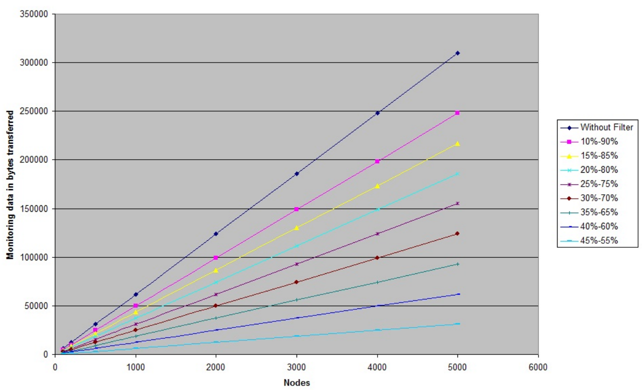

Figure 8. Evaluation to Scalability and Adaptability II

The results show that filters can assist to reduce the influence of scalability over network load. For this reason, the filtering method application in the proposed architecture can result in performance to cloud monitoring solutions. Therefore, the proposed architecture is functional to reach the scalability requirement because it can cope with a large number of agents/probes. In addition, the filtering assists to reduce the amount of monitoring data in a network. Thus, the architecture grows in a number of hosts in a way that allowing to suitable work with the scalability issues.

C. Adaptability

In adaptability evaluation, we analyze Figure 7 and Figure 8 also utilized in scalability evaluation. In this way, we show the same factors that contribute to address the scalability issues can also contribute to solving adaptability in the proposed architecture.

The Figure 7 shows that filters assists to reduce the amount of monitoring data in a network. Therefore, filters are beneficial to adaptability because the reduced amount of monitoring data in the network allows for network flows of different purposes to operate with more network resources. Thus, filters assist to reduce the workload of monitoring systems helping this type of system in order to cannot affect other activities in a cloud environment. For this reason, adaptability is influenced in a positive way by filtering.

Additionally, The Figure 8 shows that when the number of nodes grows more influence have the filters over the amount of monitoring data. Acting to reduce the amount of monitoring data. Thus, more effective is the influence of filtering to helping over adaptability issues.

The proposed architecture is based on local filters that decrease also the propagation of monitoring data over the network. The propagation is reduced because filtering is done by local filters located in local hosts. This architectural function assists to reduce the impact of scalability over adaptability. Moreover, local filters anticipates the filtering process in order to reduce the management communication among layers, decreasing the network overload. In addition, local filters improve filtering because allow to anticipate analyzes of monitoring data making faster the process of data analyzing. Thus, local filters are an architectural function that contributes to adaptability.

Finally, we can conclude that the proposed architecture is suitable to work with adaptability requirement on cloud monitoring systems because it operates in order to attend this cloud monitoring requirement.

D. Analyzes of Scalability, Adaptability, and Accuracy

The results show that cloud monitoring requirements such as scalability, accuracy, and adaptability have mutual influence among itself. In this paper, we proposed an architecture to improve scalability. Furthermore, the adaptability requirement improves also, because these two types of cloud monitoring requirements have a narrow relationship.

Scalability is an ability that a cloud monitoring system has to grow in the amount of agent/probes. On the other side, adaptability is an ability that a cloud monitoring system has to not be invasive, thereby cannot impede other activities of cloud. However, how more agents are added in a scalable cloud monitoring system, more monitoring data are generated and more influence negative has scalability over adaptability.

The proposed architecture addresses this influence using local filters for two main reasons. First, the utilization of filtering is a consolidated method in order to reduce the amount of data, in this case particularly, monitoring data. Second, the localization of filters in the proposed architecture allows to reduce the management communications among layers, decreasing the amount of monitoring data in the network.

On the other hand, the filtering used in the architecture has a negative influence over the accuracy because filters reduce the monitoring data. Thus, there are less monitoring data to be analyzed impairing the accuracy requirement. In general, the results show that accuracy resulting is minor than the exact value. For this reason, we can conclude that scalability has a negative influence over accuracy in the proposed architecture.

Finally, possibly all of cloud monitoring requirements are interrelated. In this paper, we showed an architecture where were analyzed the impact of scalability over adaptability and accuracy. These impacts can be positive or negative to each one requirement, always considering the environment and solution proposed. However, the main important point is to find out a balance among the cloud monitoring requirements.

V. Conclusions and Future Work

In this paper, we propose an architecture to analyze and evaluate cloud monitoring requirements such as scalability, accuracy, and adaptability. The architecture proposed is simplified, and based on local filters that anticipate the filtering process. The anticipate of filtering improves the cloud monitoring requirements such as scalability, and adaptability.

In this context, we can evaluate that scalability is improved because the proposed architecture is simplified and can cope with a large number of agents. On the other side, adaptability is improved because local filters allow faster the analyzes of monitoring data, and the architecture reduces management communication among layers. These affirmative is supported by an evaluation to each one of the requirements presented in section IV.

In addition, the evaluation of requirements confirms the mutual influence among scalability, adaptability, and accuracy. Moreover, the results confirmed that utilizing mild filtering could assist to find out a balance between adaptability and accuracy (*e.g.,* by filtering 20% - 80%). However, if any type of filter is applied over a set of monitoring data the accuracy is impaired and adaptability has benefits. For this reason, when a stringent filtering is applied, there are advantages to adaptability and the accuracy has a considerable loss.

It is worth mentioning that these are the first assessments about the mutual influence among scalability, adaptability, and accuracy. Thus, further evaluations regarding the correlation among these requirements are necessary to reach the exact relationship.

Finally, there are several gaps to be evaluated about cloud monitoring requirements. For example, the mutual influence among other requirements, how to fulfill a specific requirement, find out a comprehensive cloud monitoring system that meets all cloud monitoring requirements.

In this regard, future works are concentrated in accomplishing further evaluations aiming to evaluate the exact relation among scalability, adaptability, and accuracy. Moreover, to evaluate other cloud monitoring requirements, analyze the mutual influence among other requirements and the influence of scalability over other requirements.

References

[1] M. Armbrust, A. Fox, R. Griffith, A. D. Joseph, R. Katz, A. Konwinski, G. Lee, D. Patterson, A. Rabkin, I. Stoica, and M. Zaharia, "A view of cloud computing," *Commun. ACM*, vol. 53, no. 4, pp. 50–58, Apr. 2010. [Online]. Available: http://doi.acm.org/10.1145/1721654.1721672

[2] Q. Zhang, L. Cheng, and R. Boutaba, "Cloud computing: state-of-the-art and research challenges," *Journal of Internet Services and Applications*, vol. 1, no. 1, pp. 7–18, 2010.

[3] G. Rodrigues, V. Guimaraes, G. Santos, L. Tarouco, and L. Granville, "Network and services monitoring: A survey in cloud computing environments," in *Proceedings of the 11th International Conference on Networks*, ser. ICN '12. IARIA, 2012, pp. 7–13. [Online]. Available: http://www.thinkmind.org/index.php?view=articlearticleid=icn201212010123

[4] Amazon, "Amazon CloudWatch," 2013, available at: http://aws.amazon.com/en/cloudwatch/. access in: Apr 2013.

[5] S. De Chaves, R. Uriarte, and C. Westphall, "Toward an architecture for monitoring private clouds," *Communications Magazine, IEEE*, vol. 49, no. 12, pp. 130 –137, december 2011.

[6] S. Clayman, A. Galis, C. Chapman, G. Toffetti, L. Rodero-Merino, L. M. Vaquero, K. Nagin, and B. Rochwerger, "Monitoring service clouds in the future internet." in *Future Internet Assembly*, G. Tselentis, A. Galis, A. Gavras, S. Krco, V. Lotz, E. P. B. Simperl, B. Stiller, and T. Zahariadis, Eds. IOS Press, 2010, pp. 115–126. [Online]. Available: http://dblp.uni-trier.de/db/conf/fia/fia2010.html#ClaymanGCTRVNR10

[7] G. Aceto, A. Botta, W. De Donato, and A. Pescapè, "Survey cloud monitoring: A survey," *Comput. Netw.*, vol. 57, no. 9, pp. 2093–2115, Jun. 2013. [Online]. Available: http://dx.doi.org/10.1016/j.comnet.2013.04.001

[8] R. Buyya, C. S. Yeo, S. Venugopal, J. Broberg, and I. Brandic, "Cloud computing and emerging it platforms: Vision, hype, and reality for delivering computing as the 5th utility," *Future Gener. Comput. Syst.*, vol. 25, no. 6, pp. 599–616, Jun. 2009. [Online]. Available: http://dx.doi.org/10.1016/j.future.2008.12.001

[9] E. Feller, L. Rilling, and C. Morin, "Snooze: A scalable and autonomic virtual machine management framework for private clouds," in *Proceedings of the 2012 12th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (ccgrid 2012)*, ser. CCGRID '12. Washington, DC, USA: IEEE Computer Society, 2012, pp. 482–489. [Online]. Available: http://dx.doi.org/10.1109/CCGrid.2012.71

[10] J. Montes, A. Sánchez, B. Memishi, M. S. Pérez, and G. Antoniou, "Gmone: a complete approach to cloud monitoring." 2013, doi:10.1016/j.future.2013.02.011. [Online]. Available: http://www.gmrv.es/Publications/2013/MSMPA13

[11] P. Hasselmeyer and N. d'Heureuse, "Towards holistic multi-tenant monitoring for virtual data centers," in *Network Operations and Management Symposium Workshops (NOMS Wksps), 2010 IEEE/IFIP*, 2010, pp. 350–356.

[12] S. Clayman, R. Clegg, L. Mamatas, G. Pavlou, and A. Galis, "Monitoring, aggregation and filtering for efficient management of virtual networks," in *Proceedings of the 7th International Conference on Network and Services Management*, ser. CNSM '11. Laxenburg, Austria, Austria: International Federation for Information Processing, 2011, pp. 234–240. [Online]. Available: http://dl.acm.org/citation.cfm?id=2147671.2147708

[13] A. Galis, S. G. Denazis, A. Bassi, P. Giacomin, A. Berl, A. Fischer, H. de Meer, J. Srassner, S. Davy, D. F. Macedo, G. Pujolle, J. Rubio-Loyola, J. Serrat, L. Lefèvre, and A. Cheniour, "Management architecture and systems for future internet networks." in *Future Internet Assembly*, G. Tselentis, J. Domingue, A. Galis, A. Gavras, D. Hausheer, S. Krco, V. Lotz, and T. Zahariadis, Eds. IOS Press, 2009, pp. 112–122. [Online]. Available: http://dblp.uni-trier.de/db/conf/fia/fia2009.html#GalisDBGBFMSDMPRSLC09

[14] S. Ostermann, A. Iosup, N. Yigitbasi, R. Prodan, T. Fahringer, and D. Epema, "A performance analysis of ec2 cloud computing services for scientific computing," in *Cloud Computing*, ser. Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering, D. Avresky, M. Diaz, A. Bode, B. Ciciani, and E. Dekel, Eds. Springer Berlin Heidelberg, 2010, vol. 34, pp. 115–131. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-12636-99

[15] Z. Hill and M. Humphrey, "A quantitative analysis of high performance computing with amazon's ec2 infrastructure: The death of the local cluster?" in *Grid Computing, 2009 10th IEEE/ACM International Conference on*, 2009, pp. 26–33.

[16] R. Mian, P. Martin, and J. L. Vazquez-Poletti, "Provisioning data analytic workloads in a cloud," *Future Gener. Comput. Syst.*, vol. 29, no. 6, pp. 1452–1458, Aug. 2013. [Online]. Available: http://dx.doi.org/10.1016/j.future.2012.01.008

[17] Amazon, "Amazon EC 2," 2013, available at: http://aws.amazon.com/ec2/. access in: Sep 2013.

[18] M. Kutare, G. Eisenhauer, C. Wang, K. Schwan, V. Talwar, and M. Wolf, "Monalytics: online monitoring and analytics for managing large scale data centers." in *ICAC*, M. Parashar, R. J. Figueiredo, and E. Kiciman, Eds. ACM, 2010, pp. 141–150. [Online]. Available: http://dblp.uni-trier.de/db/conf/icac/icac2010.html#KutareEWSTW10

[19] C. Wang, K. Schwan, V. Talwar, G. Eisenhauer, L. Hu, and M. Wolf, "A flexible architecture integrating monitoring and analytics for managing large-scale data centers," in *Proceedings of the 8th ACM international conference on Autonomic computing*, ser. ICAC '11. New York, NY, USA: ACM, 2011, pp. 141–150. [Online]. Available: http://doi.acm.org/10.1145/1998582.1998605

[20] Nagios Enterprises, "Nagios," 2013, available at: http://www.nagios.org/. access in: Aug 2013.

[21] Cacti, "Cacti," 2013, available at: http://www.cacti.net/. accessed in: Aug 2013.

[22] D. H. Bailey, E. Barszcz, J. T. Barton, D. S. Browning, R. L. Carter, L. Dagum, R. A. Fatoohi, P. O. Frederickson, T. A. Lasinski, R. S. Schreiber, H. D. Simon, V. Venkatakrishnan, and S. K. Weeratunga, "The nas parallel benchmarks, summary and preliminary results," in *Proceedings of the 1991 ACM/IEEE conference on Supercomputing*, ser. Supercomputing '91. New York, NY, USA: ACM, 1991, pp. 158–165. [Online]. Available: http://doi.acm.org/10.1145/125826.125925

[23] P. Barham, B. Dragovic, K. Fraser, S. Hand, T. Harris, A. Ho, R. Neugebauer, I. Pratt, and A. Warfield, "Xen and the art of virtualization," in *Proceedings of the nineteenth ACM symposium on Operating systems principles*, ser. SOSP '03. New York, NY, USA: ACM, 2003, pp. 164–177. [Online]. Available: http://doi.acm.org/10.1145/945445.945462

[24] W. Stallings, *SNMP,SNMPV2,SNMPv3,and RMON 1 and 2*, 3rd ed. Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc., 1998.