2015 The 5th International Workshop on Computer Science and Engineering
doi: 10.18178/wcse.2015.04.065

Approaches to the Effective Use of Limited Computing Resources in Multimedia Applications in the Educational Institutions

Irina Bolodurina, Denis Parfenov

Department of Applied mathematics, Orenburg State University, Orenburg, 460018, Russia

Keywords: Cloud computing, cloud system, computing node, computing resource, highload information systems, load balancing, multimedia services, user request, virtual machine, virtual resource.

Abstract. Existing approaches to the use of computing resources is not efficiently for educational institutions. Modern multimedia services require significant computing power, which are not always available. In our investigation, we introduce an approach that allows more efficient use of limited resources by dynamically scheduling the distribution of data streams at several levels: between the physical computing nodes, virtual machines, and multimedia applications with use cloud computing.

Introduction

The Information flows between computing nodes in local and global networks, each year has been steadily increasing. This applies not only to large data processing centres (DPC), but for locally data centres (DC), specializing in solving problems relevant in areas such as industry, economy, health and others. Universities are increasingly using their own DC to support integrated automated information systems (IAIS), provides end users with web multimedia services.

The need in more resources is one of the problems of high-loaded IAIS. It has been established that the dependence of the consumption of resources, as opposed to available volume is an exponential. Flow analysis service requests IAIS established that they have a heterogeneous structure [1]. Modern services IAIS are based on the concept of cloud computing. However, the problem of limited resources used for the cloud system remains relevant [4].

The use of virtualization and cloud computing allows consolidate multiple online services situated in virtual machines (VM). This reduces the number of physical servers. But to effectively deploy applications on VM is necessary to solve the problem of resource planning based on variable loads and service level agreement (SLA) [3]. The most flexible architecture of cloud computing is the infrastructure as a service (IaaS). This architecture allows the user to control a pool of computing resources. Cloud computing can achieve significant cost savings by increasing the density of the load [2].

However, the consolidation of computing power and reduce the overhead of the infrastructure does not remove the existing restrictions on the performance of cloud systems. For efficient resources use of cloud infrastructure necessary develop methods and algorithms allows control the main objects. For this purpose, was determined models of resource virtualization.

Model of resource virtualization of cloud systems

In ours research, we developed a model of computing resources of the cloud systems. The conception of virtualization of computing resources is constructed on the basis of abstractions representing tuples of relations interconnected elements subsets.

The cloud system can be represented as a set of interconnected objects. They include the compute nodes (*Snode)*, systems storage (*Sstg)*, network attached storage (*Snas)* and scheduling servers (*Srasp*). The cloud system allows to run on each compute node multiple instances of virtual machines. Represent as a tuple:

$$Snode_i = \{VM_{i,1}, VM_{i,2}, ..., VM_{i,k}\}, \quad (1)$$

where k is the number of virtual machines on a compute node i, $i = 1...l$.

The network attached storage includes a set of predefined virtual machines.

$$Snas_y = \{VMimg_{y,1}, VMimg_{y,2}, ..., VMimg_{y,p}\}, \quad (2)$$

where $y = 1 ... z$ (z - number of network attached storage).

The work of entire cloud system is built using the planning system and performing any operations defined by the scheduling servers.

$$Srasp = \{Rtask_1, Rtask_2, ..., Rtask_f\}, \quad (3)$$

Each of the components of the set of nodes $Shcn = \{Snode, Snas, Srasp, Sstg, VM, ...\}$, cloud systems has the following characteristics:

$$Shcn = (State, Mem, Desk, Diskn, Core, Lan), \quad (4)$$

where $State \in$ {"on","off"} state of the object cloud system;

$Mem \in N$ – size of RAM installed for the node of the cloud system;

$Disk \in N$ – amount of disk space for storage installed for the node of the cloud system;

$Diskn \in N$ – number of storage devices installed for the node of the cloud system;

$Core \in N$ – number of processor cores installed for the node of the cloud system;

$Lan \in N$ – maximum bandwidth for network adapter for the node of the cloud system;

The cloud system is a dynamic object, changing at random time t, formalizes its status as oriented graph form:

$$Shcn(t) = \{Node(t), Connect(t), App(t)\}, \quad (5)$$

where $Node(t) = \{Node_1, \ldots, Node_\lambda\}$ the node of graph are active elements included in one of the sets $Snode_i$, $Sstg_j$, $Snas_k$, $Srasp_m$;

$Connect(t) = \{Connect_1, \ldots, Connect_\nu\}$ - active connections of users to the virtualized applications;

$App(t) = \{App_1, \ldots, App_n\}$ - active instances of applications running on VM.

In our research we determined the composition of computing resources, and the structure and mechanisms of interaction links the main components of the cloud system. For optimize the mechanism for providing access to resources information system is necessary to analyze the main flows of data transferred in the cloud system.

Model of processing data flows in highload information systems based on the cloud computing

For flows analysis in our study, we used information systems of the educational institution. For analysis determined the most popular multimedia services. The research will consider distance education system (DES) consisting different interactive applications.

The research implemented tier classification sought applications:

- Level 1: The subsystem for monitoring the students' knowledge in real time;
- Level 2: The subsystem of the electronic library;
- Level 3: The subsystem of webcasts and webinars.

Information flows at each level has its own characteristics. The intensity of service flows considered information system applications requested by users depends on levels target application. In general, the intensity of service revenues and the flow of requests for each class of applications is determined by the distribution function, which describes the distribution of the following laws:

- for level 1 - Chi-square distribution;
- for level 2 - Weibull distribution;
- for level 3 - Pareto distribution.

Flows of data transmitted in to IAIS usually processed in several phases. Define the purpose of each of the phases of service requests for their location in the processing sequence. The first phase - the distribution of data flows between the IAIS resources. The second phase - the dynamic scaling

of the computing resources. The third phase - data processing applications, storage systems and databases. Full tuple of elements of cloud system represent in the form of:

$$IS = \left\{S_1^0,\ldots,S_l^0,S_1^1,\ldots,S_n^1,S_1^2,\ldots,S_m^2,S_1^3,\ldots,S_k^3S_1^4,\ldots,S_p^4\right\} \quad (6)$$

where s_i^j - *i* element of the *j* phase (S_i^0 - transmit data flows into the system, S_i^4, received data flow from the cloud);

m∈N, n∈N, k∈N - number of elements included in the system for the respective phases *f*.

Each element S_i of the information system at any time can service multiple requests from different users. In the process the query user generated data flows upstream and downstream of the elements of the system have individual characteristics that vary over time.

Designate all incoming flows of element S_i^j as X_i^j, and the output Y_i^j, where *i* - number of the element on the *j* phase service. Each thread requests can describe a set of characteristics. Suppose, for there is an element of S_i^j, l_i^j incoming streams and p_i^j leaving. Then for the incoming flow ν=1..l_i^j and output flow μ=1.. p_i^j, we introduce a set of characteristics:

$$X_i^{(j,\nu)}(t) = \left(x_{1,i}^{(j,\nu)}(t),\ldots,x_{k,i}^{(j,\nu)}(t)\right)^T \quad (7)$$

$$Y_i^{(j,\mu)}(t) = \left(y_{1,i}^{(j,\mu)}(t),\ldots,y_{k,i}^{(j,\mu)}(t)\right)^T \quad (8)$$

Consequently, all the input flows of information system, implemented using cloud computing can be represented as:

$$X^j = \bigcup_{i=0}^{n_j} X_i^j \Rightarrow X = \bigcup_{j=0}^{f} X^j \quad (9)$$

where *j* is number of the maintenance phase, n_j for *j* number of phase elements, *f* is the number of phases of service. For output flows fairly similar condition:

$$Y^j = \bigcup_{i=0}^{n_j} Y_i^j \Rightarrow Y = \bigcup_{j=0}^{f} Y^j \quad (10)$$

To effectively serves user requests, forming flows of data in the information system, there must be an unambiguous mapping of the form $R: X \to Y$.

Graph the transition between the phases can be obtained by using the function:

$$Y_e^{j-1} = R(X_i^{j,\nu}), \qquad Y_e^{j-1} \in Y \quad (11)$$

where *e* - element number belonging to phase *j*-1 and directing the flow of data ν to the element S_i^j phase *j*, ν=1..l_i^j.

Then the set of all input flows for any element S_i^j derived from the element S_i^{j-1} information system located in the previous phase, represented in the form:

$$X_i^{j,j-1} = R_j^{-1}\left[Y_i^{j-1} \cap R(X_i^j)\right] \quad (12)$$

where *j* is the phases of service.

Then effluents element S_i^j directed to the element S_i^{j+1} represented in the form:

$$Y_i^{j,j+1} = Y_i^j \cap R(X_i^{j+1}) \quad (13)$$

In real systems, the information is usually output flows may overlap and get service on the same computing node or implemented through a single instance of the application that result in the formation of internal queues each of the phases of maintenance. To describe this process it is

necessary to determine the connection data element S_i^j flows output phase j with all elements of the phase j +1. Given this set of Y^{j*} expands and takes the form:

$$Y^{j*} = \bigcup_{S_i^j} \left[Y_i^{j,0} \bigcup \left(\bigcup_{S_i^{j+1}} Y_i^{j,j+1} \right) \right] \tag{14}$$

Thus for the final phase of the service, you must also consider the transfer of user data.

For a description of intersecting streams within the phase introduce two functions:

$$X^{j,j+1} = Q_x^j(Y^{j*}) \tag{15}$$

$$Y^{j,j+1} = Q_y^j(Y^{j*}) \tag{16}$$

where $Q_x^j(Y^{j*})$ characterizes the input and output $Q_y^j(Y^{j*})$ intersecting flows directed phase j+1.

Similarly, one can define a set of input flows entering the maintenance phase. Consequently, input data flows arriving on the element S_i^j, the phase belonging to j, of all elements of the phase j-1 may be represented as:

$$X^{j*} = \bigcup_{S_i^j} \left[X_i^{j,0} \bigcup \left(\bigcup_{S_i^{j-1}} X_i^{j,j-1} \right) \right] \tag{17}$$

To describe the intersecting flows emerging from the phase we introduce two functions:

$$X^{j,j-1} = P_x^j(X^{j*}) \tag{18}$$

$$Y^{j,j-1} = P_y^j(X^{j*}) \tag{19}$$

where $P_x^j(X^{j*})$ characterizes the input and output $P_y^j(X^{j*})$ intersecting flows directed phase j -1.

Thus, the functions (14) and (18) describe the flow of data between phases of servicing of the information system disposed in the cloud.

Then data flows in the information system disposed in the cloud can be represented as:

$$Y_i^j = R^j(X_i^j) = \begin{cases} R(X_i^j), & X_i^j \in X^j \\ P_Y^j(X^{j*}), & X^{j*} \in \bigcup_{S_i^j} \left[X_i^{j,0} \bigcup \left(\bigcup_{S_i^{j-1}} X_i^{j,j-1} \right) \right] \\ Q_X^j(Y^{j*}), & Y^{j*} \in \bigcup_{S_i^j} \left[Y_i^{j,0} \bigcup \left(\bigcup_{S_i^{j+1}} Y_i^{j,j+1} \right) \right] \end{cases} \tag{20}$$

Data flows and their characteristics may change over time and should be supplemented in the description of by parameter t.

Algorithm for controlling virtual resources of the cloud system

Represented models allow determining the most appropriate computing nodes of the information system and the virtual machines with multimedia applications. The main objective is to provide the required number of nodes for uninterrupted user work. The main task of control of computing resources is the selection their number in each subsequent point time, for preparing plan. The effective planning particularly relevant when organization of access to highload information systems, because created by the load on the services may varied within wide limits in a relatively short time intervals.

As distinct from other information systems the stream of users requests in the educational environment, possible to predict through the organization of subscriptions to multimedia services. The control algorithm used for granting users access to virtual information resources, working in two related with each other processes.

One of these processes is creating plan. During operation with dedicated resources highly loaded information system the algorithm of control collects data coming into the system requests and analyzing, classifying them according to the levels. The input data for the control algorithm are the applications of information systems, images of virtual machine, configuration of hardware and software features.

Based on this template, and data analysis given the history of connections to the computing resources of information system, located in the cloud, the control algorithm calculates the configuration required to deploy the service. For optimize the use of dedicated computing resources, the algorithm generates three variants of VMs configurations.

The first variant - provides performance headroom in the event of an unexpected increase in the number of users. The scaling factor in this case is calculated dynamically. The cloud system will be scaled within all available at the current time of computing resources. Algorithm for controlling virtual resources of the cloud system, using data about the current configuration forms the range of the number of instances.

The second variant - a fortiori provides less performance of VMs compared with a predetermined number of users. It allows to reduce the overhead of a relatively small number of users .Thus, is performed computing tasks separation between common independent nodes. It also ensures the quality of service for users using the application.

The third variant - creation using the specified in configuration user characteristics, including a fixed scaling, and a fixed number of copies of VMs that will run regardless of the number of users.

The second process was proposed algorithm for direct servicing the user requests and scaling resources during work of applications. Algorithm for controlling virtual resources and applications into account the total number of requests from each of the sources, which makes it possible to predict the load on the running applications.

The approach used in the proposed algorithm controlling virtual resources and applications, allows to take into account the physical limitations of computational resources, and organize the work adjusting the number of instances running applications.

Experimental part

For assess the effectiveness of algorithm for controlling virtual resources of the cloud system, constructed based on the presented models, with different parameters. As reference data for comparison in the experiment used standard algorithms used in the cloud system Openstack [5].

For analyze in an experiment use the data stream of requests similar to the real traffic information system. The number of concurrent requests received by the system amounted to 10,000, which is the maximum number of users.

All requests are classified into six groups, which characterize the typical behaviour of users of information. Groups of experiments of the first to third directed to service requests to the dedicated applications information systems and also concurrent background using other applications. The experiments describe the operation of the selected applications in the information system, in an excess number of concurrent requests. The experiment was one hour which corresponds to the longest period of time peak load of the system.

The results of the experiments, we can conclude a decline of 12-15% in the number of failures in service when accessing multimedia services with limited resources. In a pilot study in the OpenStack compares consumption dedicated virtual resources by the number of virtual servers for each of the subsystems. At the same time due to the optimization algorithms may release 20 to 30% of the allocated resources.

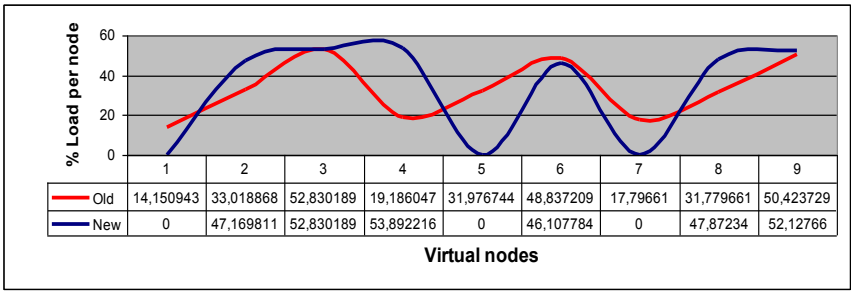

Figure 1: Graph load balancing between nodes in the cloud system

Conclusions

Thus, assessing the overall performance of the algorithm for managing virtual resources of the cloud system can get a performance boost from 12 to 15% compared with the standard, which is very effective at high intensity requests. Besides reducing the number of dedicated virtual resources can scale more efficiently scale cloud system, and provide a safety margin with a sharp increase in the intensity of use of dedicated applications.

Acknowledgements

The research work was supported by Russian Foundation for Basic Research project No. 13-07-00198 A.

References

[1] Qingjia Huang, Kai Shuang, Peng Xu, Jian Li, Xu Liu, Sen Su *Prediction-based Dynamic Resource Scheduling for Virtualized Cloud Systems* Journal of Networks, Vol 9, No 2 (2014), 375-383, Feb 2014. http://doi:10.4304/jnw.9.2.375-383

[2] S. J. E. C. I. C. Clark, K. Fraser and A. Warfield, "Live migration of virtual machines, " In Proc. NSDI, 2005.

[3] N. Bobroff, A. Kochut, and K. Beaty, "Dynamic placement of virtual machines for managing SLA violations," in Integrated Network Management, 2007. IM'07. 10th IFIP/IEEE International Symposium on. IEEE, 2007, pp. 119–128.

[4] Q. Huang, S. Su, S. Xu, J. Li, P. Xu, and K. Shuang, "Migration-based elastic consolidation scheduling in cloud data center," in Proceedings of IEEE ICDCSW 2013.