

Load balancing algorithms in cluster systems

Andrzej Imiełowski^{1,*}

¹The State Higher School of Technology and Economics in Jarosław, Poland

Abstract. The beginning of XXI century brought dynamic rise in popularity of the Internet. It was connected with the development of telecommunications technology, especially the spread of broadband access to the global network. The key issue became ensuring satisfactory access to the services available through the network. Therefore, the challenge caused the development of various clusters. The paper presents an overview of algorithms for load balancing used in modern cluster systems. Load balancing is a key issue in cluster systems. It is a method that uses multiple nodes (servers) and dynamically distributes the load between them so that no node is not overloaded. The main goal of load balancing is the optimal use of resources, thereby enhancing system performance and minimizing resource consumption. This allows for reducing operations costs of information systems and is also compatible with Green Computing. The paper focuses on analysis of existing algorithms in terms of load balancing and comparing them on basis of various quality indicators, such as throughput, reliability, energy saving, performance, scalability, etc. The paper takes into account the specificities of different solutions, from the classic cluster through systems of blade servers with virtualization to cloud systems.

1 Introduction

The modern world is more and more dependent on information systems. It is due to both the rapid development of the Internet and services made available through it and with the common application of solutions for information processing in all spheres of life from business through administration to private information resources (documents, photos, correspondence, etc.). The widespread use of IT systems implies ever greater demands on the quality of services that are provided by these systems. An important parameter determining the quality of IT solutions is their efficiency. This factor has a significant impact on the comfort of use and in extreme situations may result in the inability to use a given solution. For the above reasons, actions aimed at improving the efficiency of IT systems are of great importance for the development of IT applications. In recent years, it has become common to use cluster systems to support IT systems requiring high performance. At the same time, solutions that use virtualization as a working environment of server systems have become very popular (for many reasons). The above two factors, the development of cluster systems and the widespread use of virtualization have enabled the popularization of services related to cloud systems (cloud computing). This article focuses

* Corresponding author: andrzej.imielowski@pwste.edu.pl

on the development of load balancing algorithms in multi-server systems from classic clusters to cloud systems.

2 Architecture of cluster systems

Research and development works aimed at improving the efficiency of IT systems are multidirectional. These works cover all areas that can affect performance. This applies to the development of the software itself, both application and system. The second direction of work concerns the development of hardware solutions that increase efficiency. In connection with the development of equipment, the important issue is, in addition to performance, also increasing the reliability parameters of computer systems.

The proposed solutions are aimed in the following directions:

- dedicated multiprocessor solutions (generally aimed at improving performance through new hardware solutions within a single computer)
- solutions for improving the parameters of both application software and operating systems
- distributed solutions - systems based on multiple computers.

Independently from work related to the development of application software, the work on the scalability of the applied solutions depending on the current needs has the decisive influence on improving the efficiency of IT systems [1]. Scalability is most often achieved through the use of multi-server systems. The classification of methods to improve the performance of scalable systems is shown in Figure 1.

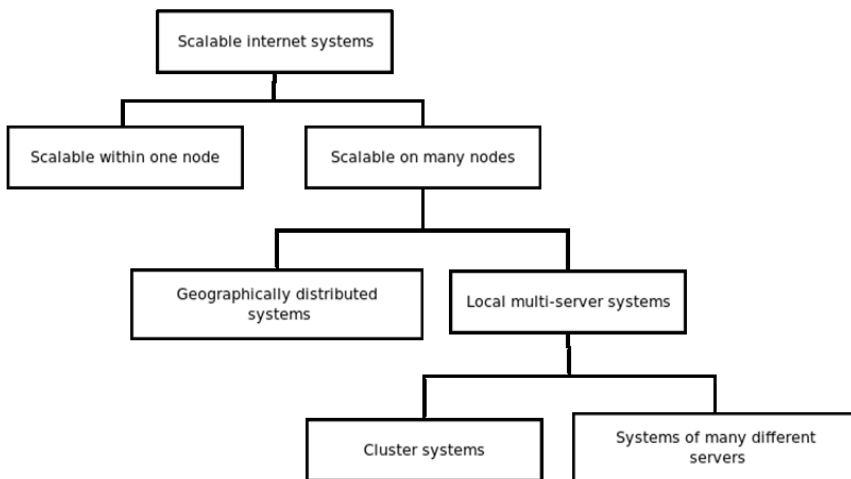


Fig. 1 Methods of improving internet applications performance

Scalable solutions based on the use of multiple servers can be divided into two types:

- systems consisting of servers located in different locations, but functioning in such a way that for the client they form one system
- systems consisting of multiple servers located in one location:
 - cluster systems consisting of multiple servers visible outside as one system (common IP address)
 - systems consisting of multiple servers between which system tasks are distributed (the client can distinguish between individual servers, they have different IP addresses)

This article focuses on the solutions from the second group, i.e. on local cluster systems. The architecture of the classic server cluster is shown in Figure 2. In the simplest case, such a cluster consists of a number of identical servers and a dedicated device called dispatcher

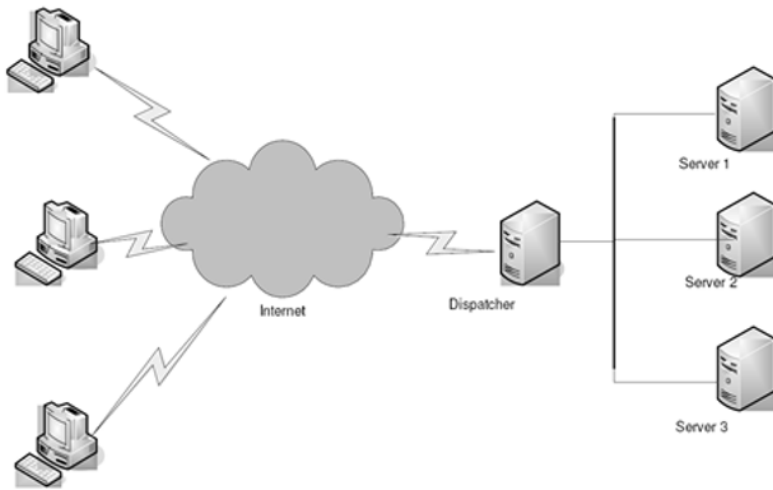


Fig. 2 Architecture of cluster

or load balancer. Cluster switch is a term sometimes used in Polish, but it does not occur too often. Dispatcher has the task of distributing tasks between servers constituting cluster nodes. In this solution, the load balancer can be a dedicated network device operating in layer 3 of the OSI network model. Devices of this type are provided by most manufacturers of network hardware. The role of the dispatcher can also be performed by a dedicated server included in the cluster - such solutions are provided by system software producers. They are available for both Windows (Windows Azure) operating systems and most UNIX / Linux operating systems. Yet another group of solutions is software for a node managing a cluster at the level of application software, e.g. managing the database system or web servers, also in this case the software producers provide such solutions (MS SQL Failover Cluster, MySQL Cluster, Oracle RAC).

Another approach allowing to increase the system's efficiency is to split the system's functionality into a larger number of servers. Figure 3 shows the idea of this solution. In contrast to the solution described above, in this case it is necessary to adapt the application

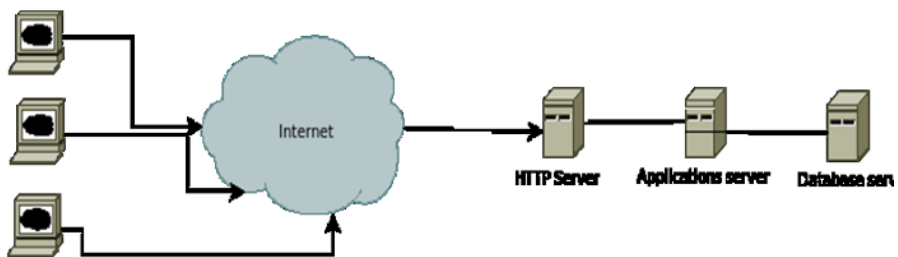


Fig. 3 Cluster with the division of server functions

software to such an organization of task division. For typical application solutions, this is

possible without any special restrictions. The figure presents cluster organizations for web applications, but it is often used in other areas, e.g. for highly available GIS applications. Often for applications requiring high performance, the two above solutions are combined, i.e. after a functional division, a cluster with duplicate nodes is used as part of a given function. The further part of the article focuses on solutions concerning clusters with nodes without division of functions.

3 Load balancing algorithms in classical cluster systems

Work of task switches in the cluster has a significant impact on system performance. The basic tasks that are set before the cluster switch are the following:

- maximizing system efficiency
- efficient use of resources
- reducing energy consumption in a case of low load
- reducing costs of use of resources

While maximizing efficiency is obviously the main task, the remaining ones listed above are the result of high cost-consumption of cluster solutions not only at the level of investment but also ongoing maintenance. Ensuring proper working conditions of the cluster due to the power quality and climatic conditions of the system is very expensive. In addition, there is a factor related to ecology, which is called Green Computing in recent years.

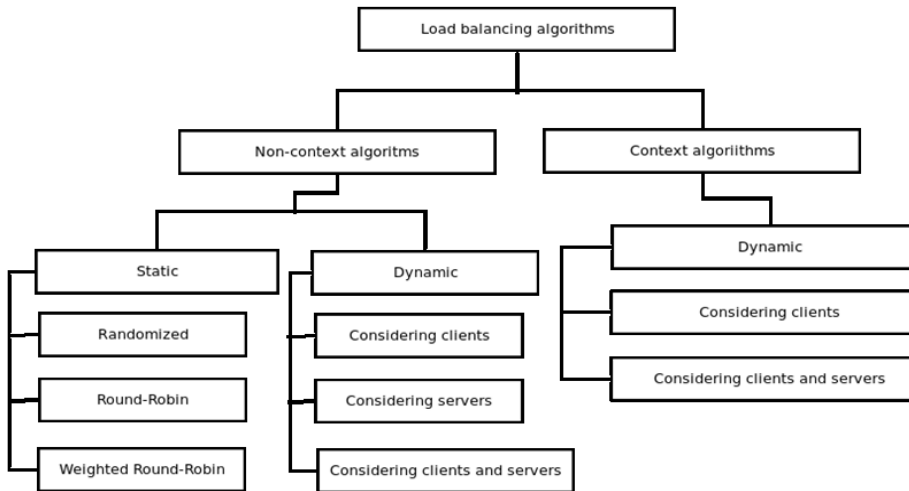


Fig. 4 Systematic of load balancing algorithms

The main task performed by a switch is to distribute loads of individual servers in a way to obtain the best performance indicators of the entire system. Dispatchers work on the basis of various task allocation algorithms. The division of task distribution algorithms in cluster systems is shown in Figure 4. For several years, works have been underway related to the development of algorithms allowing for better alignment of tasks to dynamically changing cluster working conditions such as server load, relative number of requests incoming to the system and others. Task distribution algorithms must meet one basic condition - they must have very low complexity, because the decision to choose a server must be made in real time. Two main types of algorithms are distinguished:

- contextual - taking into account the content of queries
- context-free - not taking into account the content of queries.

Among the context-free algorithms we distinguish:

- static: algorithms that operate independently of external factors, they do not use any information about the system state
- dynamic: algorithms that take into account information about the system state in their operation

Typical and most commonly seen in practice static algorithms are the algorithm that randomly assigns tasks to servers and the Round Robin algorithm [2]. The typical Round Robin algorithm is often extended to a form that takes into account the performance of individual cluster servers (in a situation where they are different), the so-called Weighted Round Robin (WRR). In this case, the servers, depending on their performance, are assigned appropriate weights that affect the frequency of task allocation.

Dynamic contextless algorithms can be divided into:

- taking into account clients
- taking into account the state of servers
- taking into account both clients and the state of servers.

Dynamic algorithms usually take into account the state of servers. This is due to the fact that they work in OSI layer 3 and 4 switches [3]. The result is that the only information about the clients that the switch has is those included in the TCP / IP packet headers - the address and port to which the connection is directed. In some cases, this information is used to divide clients into groups and to assign them specific resources within the cluster. More information is available about the status of the cluster, although the information about the load of individual servers is not directly available, but the switch has information about the status and number of active connections. Another method of determining the status of individual servers in the cluster is periodically checking by the switch the response times of individual servers and directing new connections to the servers with the shortest response time. A common solution is to modify the WRR algorithm by updating the weights of individual servers with the number of connections or response times taken into account.

A separate group of algorithms are contextual algorithms operating in OSI layer 7 switches [4]. These switches analyze the content of HTTP requests, which allows for:

- increasing the efficiency of cache usage by directing tasks with the same content to servers that have handled similar requests recently
- using specialized servers, e.g. for delivering video content
- partitioning server content and distributing functions within the service

4 Architecture of cluster systems with virtualization

In the last dozen or so years we have been observing the intensive development of server systems based on virtualization. It is connected with many other new solutions regarding both hardware and software solutions. The key from the point of view of this article is the combination of server system virtualization technologies with cloud solutions. Virtualization is a technique that allows you to separate the software layer (at the application or operating system level) from the hardware layer. This allows you to freely manage hardware resources and assign them to tasks depending on your needs. The simplest solution from this point of view is the static allocation of resources to individual virtual machines based on the forecasted load. This type of virtual system organization is found mainly in smaller systems with a limited and known number of users [5]. Possible modifications to the allocation of resources can be made relatively rarely based on the observation of the system load. In systems with greater complexity, solutions allowing for dynamic allocation of resources are used, this will be the subject of the further part of the article.

Solutions providing mechanisms for virtualization vary on many levels, but the basic idea and principles of operation are similar. The main virtualization solutions are currently:

- VMware vSphere
- Citrix XenServer
- Microsoft Azure / Hyper-V

A detailed description of solutions used in server virtualization is not the subject of this article, but it is necessary to present the general principle of operation of such a system. An example of the VMware vSphere solution has been used for this purpose.

The cluster system architecture with the use of virtualization is based on the use of a dedicated server to manage the system. The system manager (different terminology is used in different solutions) is to supervise the operation of virtual systems and direct service requests to selected virtual machines. The physical servers in such a system are managed by specialized software that provides virtualized resources (in the case of VMware ESX software). The resources of individual physical servers are allocated to virtual systems, one physical machine can be part of a larger virtual machine, but it can also host multiple virtual systems. The whole cluster can be treated as a common pool of virtual resources that can be allocated depending on the needs. Modern solutions allow for dynamic allocation of resources depending on the current load, it requires the use of advanced mechanisms that allow the migration of virtual systems between physical servers. Another mechanism that improves the efficiency of the virtual cluster is the ability to temporarily freeze the

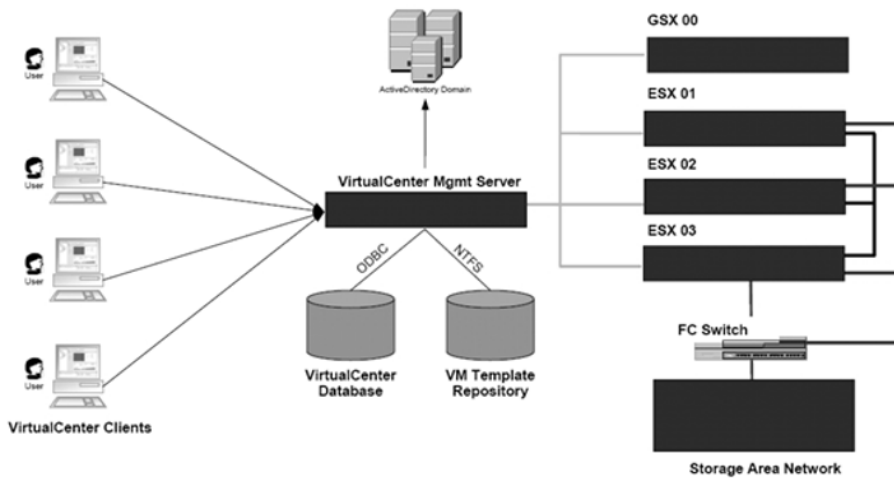


Fig. 5 Architecture of cluster with virtualization [6]

operation of unloaded systems and recreate them as needed.

5 Load balancing algorithms in cloud systems

The previous chapter presents the basic mechanisms occurring in a cluster of servers with virtualization applied. All of these mechanisms have a negative impact on system performance. It should be noted that the main benefits that are obtained from virtualization are [7]:

- consolidation of resources
- increasing the efficiency of hardware resources utilization
- centralized management

- increased resilience to critical situations
- improvement and reduction of failure removal costs

The above benefits are unquestionable, however they also cause an additional system load. Therefore, balancing the burdens in such systems becomes so important. Chapter 4 presents algorithms for load balancing in classical cluster systems. The solutions for clusters mainly concerned static context-free solutions. Dynamic contextless algorithms, in the case of classical clusters, can take into account the state of the system usually only to a small extent. This is due to the lack of information about the status of individual servers. This situation changes fundamentally for clusters with virtualization. The correct operation of the virtual system requires ongoing monitoring of the status of its components. This opens up wide possibilities of using this information to balance loads in the system. The tasks set before the balancing mechanisms will also be greater. In addition to the basic task of separating tasks between physical servers included in virtual machines, the load balancing subsystem should decide on the distribution of virtual resources within the cluster, including the deciding on migrations. Due to the above, research into algorithms optimizing the efficiency of such systems has been dynamically developing in recent years.

The tasks that are placed before the load balancing subsystem [8]:

- resources are readily available (at the desired time)
- cluster resources are used efficiently (taking loads into account)
- energy consumption is minimized (particularly in the case of a small load)
- resource utilization cost is reduced

In connection with the above tasks, the algorithms used in load balancing vary widely and are used depending on the configuration of a cluster system. Therefore, several types of load balancing algorithms can be distinguished.

- in a static environment: algorithms similar to those used in classical clusters (described in Chapter 4) are used, mainly the Round Robin algorithm with different additional weights available to the cluster management software
- in a dynamic environment, three types of algorithms are distinguished depending on which node is responsible for load balancing:
 - Centralized Load Balancing - in this case, one designated node is responsible for all tasks related to load balancing, it reduces the time necessary to make decisions but also causes a higher load on the node and reduces the system's resistance to failures
 - Distributed Load Balancing - in this case all nodes store the information needed to make decisions, which allows to ensure a high level of resistance to failures
 - Hierarchical Load Balancing - in this case to manage tasks a hierarchical structure of nodes in the cluster is used, decision making is spread over several levels (a three-level structure is most common)

Regardless of the above-described systematics, load balancing mechanisms use various kinds of algorithms. Artificial intelligence methods are increasingly in use. Table 1 shows the comparison of algorithms and their use.

Table 1 presents a comparison of algorithms that are used in various types of cloud systems. It is not the purpose of this article to provide detailed information about individual algorithms, information about them is widely available in the literature. The list above has been cited mainly to demonstrate the huge variety of currently proposed solutions.

6 Summary

The article presents basic information on issues related to load balancing in cluster systems. The tasks and limitations applicable to the algorithms in use are presented. Due to the popularization of solutions based on virtualization, there was a significant development of load balancing mechanisms, their role and their tasks increased significantly. Currently, mechanisms that come from various fields of computer science are used to build effective solutions that will allow meet the requirements. Optimization mechanisms using artificial intelligence has gained very much importance in the development of load balancing in cluster systems.

References

1. V. Cardellini, E. Casalicchio, M. Colajanni and P. S. Yu, "The state of the art in locally distributed web-server systems," ACM, (2002)
2. H. Bryhni, E. Klovning and O. Kure, "A comparison of load balancing techniques for scalable web servers.," Network, IEEE, (2000)
3. M. Colajanni and E. Casalicchio, "Scalable web clusters with static and dynamic contents.," in In Cluster Computing, 2000. Proceedings. IEEE International Conference on, (2000)
4. K. Gilly, N. Thomas, C. Juiz i R. Puigjaner, „Scalable qos contentaware load balancing algorithm for a web switch based on classical policies.,” w Advanced Information Networking and Applications, 2008. AINA, (2008)
5. Q. C. Zhang and R. Luand Boutaba, "Cloud computing: state-of-the-art and research challenges," Journal of Internet Services and Applications, pp. 7--18, (2010)
6. www.vmware.com
7. M. Xu i W. Tian, „An Online Load Balancing Algorithm for Virtual Machine Allocation with Fixed Process Intervals,” Journal of Information & Computational Science, p. 989–1001, 10 February (2014)
8. P. A. Pattanaik, S. Roy i P. K. Pattnaik, „Performance study of some dynamic load balancing algorithms in cloud computing environment,” Signal Processing and Integrated Networks, (2015)
9. M. Katyal and A. Mishra, "A Comparative Study of Load Balancing Algorithms in Cloud Computing Environment," International Journal of Distributed and Cloud Computing, (2013)