ISSN: 0975-766X
CODEN: IJPTFI
Available Online through *Research Article*
www.ijptonline.com

AN EFFECTIVE LOAD BALANCING APPROACH FOR HANDLING BIG DATA

[1]Akshit Bhandari, [2]Lavish Chhatwani[*], [3]Shivali Chauhan, Prof. [4]SenthilKumar N
1,2,3,4 School of Information Technology & Engineering,VIT University, Vellore, India.
Email: lavishchhatwani95@gmail.com

Received on 25-10-2016 *Accepted on 02-11-2016*

Abstract

In this paper, we have studied about load balancing like-Why Load Balancing is necessary? What are the algorithm we are using for load balancing, Comparison of algorithm ,and Which algorithm is best among this four algorithms. This four algorithms are: - Round Robin, Min Min, Max Min, Honey Bee Load Balancing :- Load means amount of workload across the servers, And Balance means how we manage this load. Load Balancing is a process of distributing workload across multiple computers or computer cluster with the help of load balancing we can do Efficient use of resources. Easy to maintain workload etc. Load balancing is mostly used in cloud computing, The four main services of cloud computing are-SAAS: Software as a Service, PAAS: Platform as a Service, IAAS: Infrastructure as a Service, HAAS: Hardware as a Service. We create a multiple Algorithm for load balancing.

1. Introduction

Big data is a collection of large number of datasets that cannot be processed using traditional/normal computing techniques. Bigdata is not a single technique or a tool .it involves numbers of areas of business and technology. What does come Under Big data:-Big data involves the data produced by different-2 devices and applications. These below are some of the fields that come under the Big Data.

1.1 Cloud computing: Distributed systems have seen tremendous advancement in performing various operations and cloud has been efficacious to provide a flexible and scalable approach. It isan extensive distributed computing standard & is widely adopted by organisations. The abstracted, virtualized, computing power, storage, platforms and services of cloud are delivered on demand to the users over the Internet. Google, Amazon, IBM and many others have developed their own cloud platforms which manage multiple nodes and heterogeneous applications. With the massive growth in recent years, cloud computing still poses challenges to researcher(see Table 1).

1.2 Load Balancing: Load balancing is a cloud computing method to distributed workload across multiple computers or a computer cluster, network links, central processing units, disk drives, or other resource. Load balancing is the most effective way to solve the above problem in a cloud computing infrastructure, which ensures that services are delivered transparently regardless of the physical implementation and location within the "cloud" [2]. Load balancing in cloud computing provides an efficient solution to various issues pertaining to cloud computing environment set-up and usage. Load balancing must take into account two major tasks, one is the resource provisioning or resource allocation and other is task scheduling in distributed environment [3].

Table 1: Big data references and its associations.

Black Box Data	It's a component of helicopter, airplanes, & jets, etc. It captures voices of flight crew, recordings the microphones and earphones, & performance information of the aircraft.
Social Media Data	Social media Data like Facebook & Twitter who hold information and the views posted by millions of people across the world.
Stock Exchange Data	In this topic data comes who's related the share market The stock exchange data holds information about the 'buying' and 'selling's' decisions made on a share of different-2 companies made by the customers.
Power Grid Data	This part to relate to the power .It holds information consumed by a particular node with respect to a base station.
Transport Data	Transport data mainly includes models, capacities, distances and availability of a vehicle.
Search Engine Data	Its related lots of data from different databases.
Volume of data	Volume refers to large amount of data. Volume of data stored in enterprise repositories have grown from megabytes and gigabytes to petabytes.
Variety of data	Different-2 types of data and sources of data. Data variety shows from structured and legacy data stored in enterprises repositories to type of data like unstructured, semi structured, audio, video, XML etc.
Velocity of data	It refers to speed of data processing. For time-sensitive processes such as catching fraud, big data must have used as it streams into your enterprise in order to maximize its value.

2. Algorithmic Approaches to handle Big Data

2.1 Round Robin: Round Robin is undoubtedly the most widely used algorithm. It's easy to implement and easy to understand. Here's how it works. Let's say you have 2 servers waiting for requests behind your load balancer. Once the first request arrives, the load balancer will forward that request to the 1st server. When the 2nd request arrives (presumably from a different client), that request will then be forwarded to the 2nd server.Because the 2ndserver is the last in this cluster, the next request (i.e., the 3rd) will be forwarded back to the 1st server, the 4th request back to the 2nd server, and so on, in a cyclical fashion (see Figure 1).

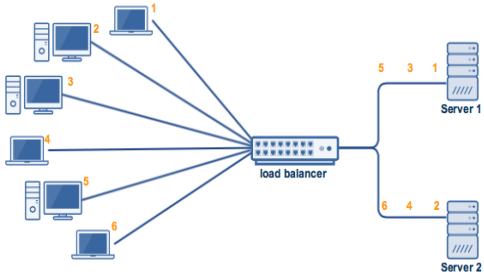

Figure 1: Round Robin Representation.

As you can see, the method is very simple. However, it won't do well in certain scenarios. For example, what if Server 1 had more CPU, RAM, and other specs compared to Server 2? Server 1 should be able to handle a higher workload than Server 2, right?Unfortunately, a load balancer running on a round robin algorithm won't be able to treat the two servers accordingly. In spite of the two servers' disproportionate capacities, the load balancer will still distribute requests equally. As a result, Server 2 can get overloaded faster and probably even go down. You wouldn't want that to happen. The Round Robin algorithm is best for clusters consisting of servers with identical specs. For other situations, you might want to look at other algorithms, like the ones below.

2.2 Weighted Round Robin: For the 2nd scenario mentioned above, i.e., Server 1 having higher specs than Server 2, you might prefer an algorithm that assigns more requests to the server with a higher capability of handling greater load. One such algorithm is the Weighted Round Robin. The Weighted Round Robin is similar to the Round Robin in a sense that the manner by which requests are assigned to the nodes is still cyclical, albeit with a twist (see Figure 2). The node with the higher specs will be apportioned a greater number of requests. But how would the load balancer know which node has a higher capacity? Simple. You tell it beforehand. Basically, when you set up the load balancer, you assign "weights" to each node. The node with the higher specs should of course be given the higher weight.You usually specifying weights in proportion to actual capacities. So, for example, if Server 1's capacity is 5x more than Server 2's, then you can assign it a weight of 5 and Server 2 a weight of 1.

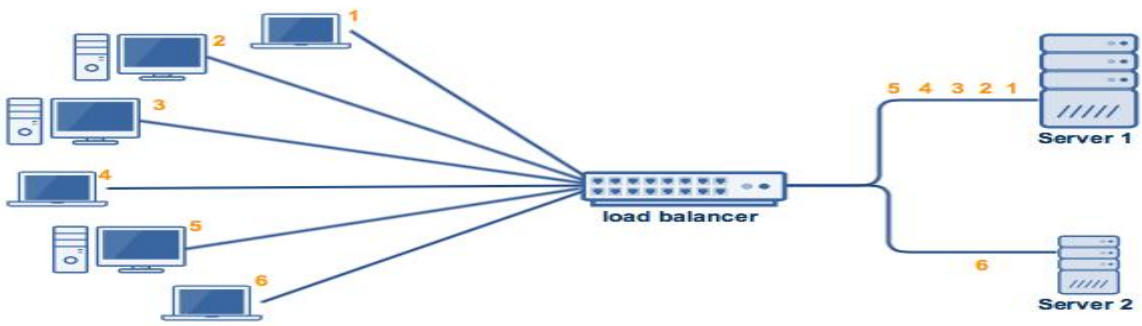

Figure 2: Weighted Round Robin Representation.

So when clients start coming in, the first 5 will be assigned to node 1 and the 6th to node 2. If more clients come in, the same sequence will be followed. That is, the 7th, 8th, 9th, 10, and 11th will all go to Server1, and the 12th to Server 2, and so on. Capacity isn't the only basis for choosing the Weighted Round Robin (WRR) algorithm. Sometimes, you'll want to use it if say you want one server to get a substantially lower number of connections than an equally capable server for the reason that the first server is running business-critical applications and you don't want it to be easily overloaded.

2.3 Least Connections: There can be instances when, even if two servers in a cluster have exactly the same specs (see first example/figure), one server can still get overloaded considerably faster than the other. One possible reason would be because clients connecting to Server 2 stay connected much longer than those connecting to Server 1(see Figure 3).This can cause the total current connections in Server 2 to pile up, while those of Server 1 (with clients connecting and disconnecting over shorter times) would virtually remain the same. As a result, Server 2's resources can run out faster. This is depicted below, wherein clients 1 and 3 already disconnect, while 2, 4, 5, and 6 are still connected.

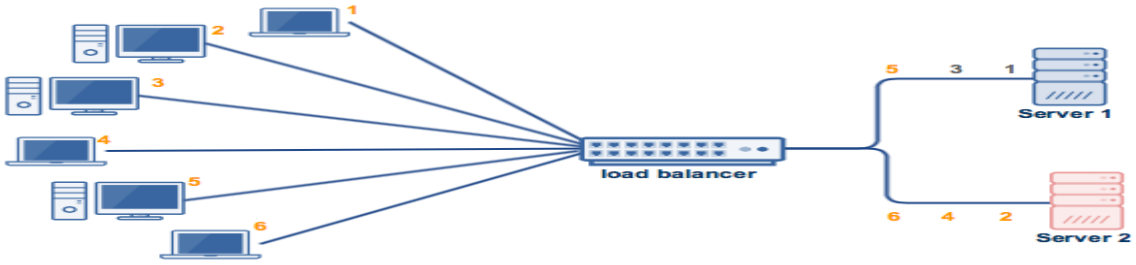

Figure 3: Least connection Representation.

In situations like this, the Least Connections algorithm would be a better fit. This algorithm takes into consideration the number of current connections each server has. When a client attempts to connect, the load balancer will try to determine which server has the least number of connections and then assign the new connection to that server.So if say (continuing our last example), client 6 attempts to connect after 1 and 3 have already disconnected but 2 and 4 are still connected, the load balancer will assign client 6 to Server 1 instead of Server 2.

2.4 Weighted Least Connections: The Weighted Least Connections algorithm does to Least Connections what Weighted Round Robin does to Round Robin. That is, it introduces a "weight" component based on the respective capacities of each server (see Figure 4). Just like in the Weighted Round Robin, you'll have to specify each server's "weight" beforehand. A load balancer that implements the Weighted Least Connections algorithm now takes into consideration two things: the weights/capacities of each server AND the current number of clients currently

connected to each server. Random:-As its name implies, this algorithm matches clients and servers by random, i.e. using an underlying random number generator. In cases wherein the load balancer receives a large number of requests, a Random algorithm will be able to distribute the requests evenly to the nodes. So like Round Robin, the Random algorithm is sufficient for clusters consisting of nodes with similar configurations (CPU, RAM, etc.)

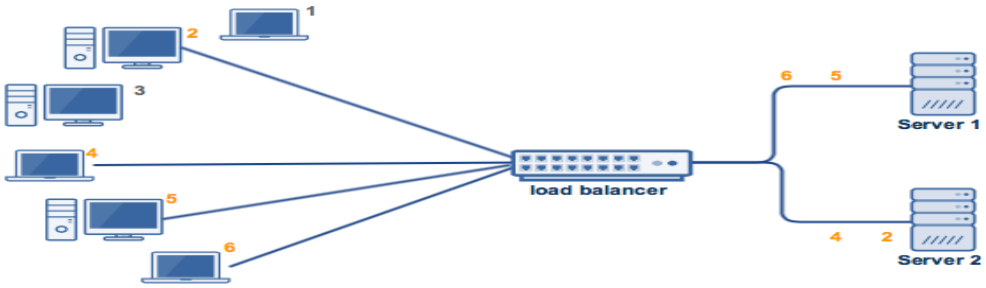

Figure 4: Weighted Least connection Representation.

2.5 Min Min Load balancing Algorithm: In this algorithm the cloud manager give priority to the job having minimum execution time by assigning them to the processor, the job having maximum execution time has to wait for the first job to be completed until all task are assigned in the processor the assigned task are updated in the processor and the task is removed from the waiting queue .The Drawback of the algorithm is that the task which is having maximum time that cannot be completed ever if there are task in the queue which are having minimum time ,that condition is known as Starvation.

2.6 Min Max Load Balancing Algorithm: In this algorithm the cloud manager give priority to the job having maximum execution time by assigning them to the processor, the job having minimum execution time has to wait for the first job to be completed until all task are assigned in the processor the assigned task are updated in the processor and the task is removed from the waiting queue.

2.7 Honey Bee Load Balancing Algorithm: Viable execution of load adjusting will assemble cloud computing more practical and it conjointly enhances client fulfilment. Inside the arranged system, a Honey bee scrounge strategy is utilized for undertaking portion and payload adjusting. When task are distributed to the VMs current load is ascertained. On the off chance that the VM gets to be over-load the errand is exchanged to the area VM whose heap worth is underneath limit. Honey bee scrounge system utilizes suburbanized load levelling strategy and task exchange are dispensed on the fly.

3. Comparison of Algorithm: We compare the algorithm on the basis of four Keywords: Throughput, Response Time , Availability , Scalability , Robustness , Starvation (see Table 2).

Throughput: The measure of material or things going through a framework or process.

Response Time: The amount of time which is Required by the server to response the client request.

Availability: The Number of available Recourses to perform the tasks

Scalability: Scalability is the ability of a framework, system, or procedure to handle a developing measure of work, or its capability to be amplified with a specific end goal to oblige that development.

Robustness: In computer science, **robustness** is the ability of a computer system to cope with errors during execution and cope with erroneous input.

Starvation: when a particular process is not able to execute that condition in known as starvation.

Table 2: Comparative Analysis of Various Big Data Algorithms.

	Throughput	**Response Time**	**Availability**	**Scalability**	**Robustness**	**Starvation**
Round Robin	**Yes**	**Yes**	**Yes**	**No**	**Yes**	**NO**
Min Min	**Yes**	**Yes**	**Yes**	**NO**	**Yes**	**Yes**
Min Max	**Yes**	**Yes**	**Yes**	**NO**	**Yes**	**Yes**
Honey Bee	**Yes**	**NO**	**NO**	**Yes**	**No**	**No**

4. Conclusion:

This Paper is based on cloud computing Technology. In this paper we explain the four load balancing algorithms. This Algorithm are Round Robin ,Min Min,Min Max, Honey Bee.We differentiate this four algorithm on the basis of this five keywords: Throughput , Response Time , Availability , Robustness , Starvation , According to this paper we can find which algorithm is best among of four. As a conclusion we find that Round Robin is the best algorithm among this four just because in round robin algorithm the Throughput ,Response Time ,Availability is better than other algorithm, and There is no Starvation Condition in this algorithm. The Worst Algorithm is Min Min load Balancing Algorithm among this four algorithms because there is starvation condition.

5. References

1. Suguna R, DivyaMohandass, Ranjani R, "A Novel Approach For Dynamic Cloud Partitioning And Load Balancing In Cloud 0Computing Environment", Department of CSE,SKR Engineering college, ISSN: 1992-8645.

2. RenGao and et al, "Dynamic Load Balancing Strategy for Cloud Computing with Ant Colony Optimization" School of Information Engineering, Hubei University of Economics, China, Future Internet 2015, 7(4), Page No. 465-483
3. MayankaKatyal*, Atul Mishra , "A Comparative Study of Load Balancing Algorithms in Cloud Computing Environment", Published online in www.publishingindia.com
4. Niranjan G. Shivaratri, Phillip Krueger, and MukeshSinghal"Load Distributing for Locally Distributed Systems", volume: 25, Issue:12 ,ISSN:0018-9162,Ohio State University.
5. Rajwinder Kaur and Pawan Luthra on "Load Balancing in Cloud System using Max Min and Min-Min Algorithm", International Journal of Computer Applications (0975 – 8887) National Conference on Emerging Trends in Computer Technology (NCETCT-2014).
6. Dharmesh Kashyap and Jaydeep Viradiya on "A Survey Of Various Load Balancing Algorithms In Cloud Computing",INTERNATIONAL JOURNAL OF SCIENTIFIC & TECHNOLOGY RESEARCH VOLUME 3, ISSUE 11, NOVEMBER 2014.ISSN 2277-8616.
7. Rajwinder Kaur and Pawan Luthra on "Load Balancing in Cloud Computing", Proc. of Int. Conf. on Recent Trends in Information, Telecommunication and Computing, ITC
8. Tushar Desai and Jignesh Prajapati on "A Survey Of Various Load Balancing Techniques And Challenges In Cloud Computing", INTERNATIONAL JOURNAL OF SCIENTIFIC & TECHNOLOGY RESEARCH VOLUME 2, ISSUE 11, NOVEMBER 2013,ISSN 2277-8616
9. N. J. Kansal and I. Chana, "Existing load balancing techniques in cloud computing: A system atic review.," Journal of Information Systems & Communication, vol. 3, no. 1, 2012.
10. AlexandruIosup, S. Ostermann, M. N. Yigitbasi, R. Prodan, T. Fahringer, and D. H. Epema, on "Performance analysis of cloud computing services for many-tasks scientific computing," Parallel and Distributed Systems, IEEE Transactions on, vol. 22, no. 6, pp. 931–945, 2011
11. Kushwah, V.S. et al. (2014), "A Survey on Various Fault Tolerant Approaches for Cloud Environment During Load Balancing". Inter. Journal of Computer Networking, Wireless & Mobile Communications, ISSN (Print): 2250-1568; E-ISSN: 22789448, Vol. 4, Issue 6, December 2014, pp. 25-34.