Unit 5 Server Solutions

**Corey Crooks**

**Purdue University Global**

**IT332 – Principles of Information Systems Architecture**

**Jonathan Abramson**

**August 9th, 2023**

Hello, ABC Widgets. It has come to my attention that you are looking to explore the possibility of computer clustering. This is a highly practiced discipline in the technological landscape, and there are some important factors to discuss and learn. This could be a very rewarding prospect if all important avenues are thoroughly examined before the required infrastructure has been committed to. Allow me to shed some light on the topic and help eliminate as much of the confusion surrounding it as I can.

To understand computer clustering, let’s break down the fundamentals in order to work upwards in logic. In a standard system, there are a number of different mechanisms all working at their most efficient to get a series of tasks done in time. If you open up a word processor to write a memo for important concepts in your technology conference, your CPU might need to run a series of checks and computes, along with communicating with your RAM, Hard Drive, GPU and peripheral devices in order to display what you want when you want it (The University of Rhode Island, n.d.). There are hundreds of thousands, if not millions, of these communications going on in a standard desktop computer nowadays, and that number is only growing with time as applications get more and more demanding. At a certain point, your system cannot keep up with the demand of the workload put in front of it. In this case, tackling things head-on may prove unfruitful. To up the efficiency of these tasks, a logic structure was developed by the name of multithreading. Essentially, this allows a compatible CPU to multitask while trying to work through a program (Amaratunga, 2000).In your CPU, there may be a number of different cores. These cores are capable of their own operations independent from other cores and their computations. Because of this, multitasking can be extremely effective. Without such a system, it could be that a certain rigorous computation might take up four of the sixteen cores available in a CPU, and the other twelve cores went unutilized. Multitasking comes in to make sure that if there are cores that aren’t being utilized, new logical computations can be directed to them to ensure full saturation where possible, and speed up the overall time it takes to get done with anything you require. All of these smaller components are working in tandem with one another to ensure that the overall checklist gets done as fast as it possibly can.

To the untrained eye, this talk about individual systems may seem unrelated to the topic at hand. Afterall, it is about computer *clusters* instead of just individual systems. But instead, you may already have noticed where this theme kicks in. Inside a CPU, there may be multiple little components that all work together. You can apply the same thing on a much bigger scale to computer clustering. In this idea, multiple different systems make up the cores of this CPU so that each system is utilizing its full bandwidth of power to deliver what it can produce to the problem at hand (Princeton Research Computing, 2023). This can be done with multiple different systems in order to work in tandem as described with CPU cores in order to split up a large and complex logic computation into smaller, more manageable chunks for each system to process and deliver the output.

This can be done by setting up a communication structure within your environment. Usually with ethernet or Fiber Optic cabling, computers can be connected to a centralized server. This server will receive the problem, and provide instructions to each of the systems in the circuit. These systems will carry out their tasks, and deliver their result. When the result is delivered, the server can wait for the rest of the auxiliary systems to compile their outputs as well, and then put it together to form the final output[REFERENCE NEEDED]. This can go even deeper, however. It is entirely possible that a company may take a more modular approach, and offer multiple possible servers instead of just one. In a chain of servers, there are algorithms in place to route traffic efficiently so that the load is shared between active servers. This ensures that no one server is completely burdened by traffic that could otherwise be shared to reduce compute times. This is referred to as High Availability Load Balancing (RStudio Connect, n.d.).This is a particularly important concept, as it isn’t necessarily vital to the operations, but will help the overall stability of the environment to create an effective workload in all connected systems.

With all these algorithms and hardware, you’re going to need somewhere to put it all. There are actually a few different ways of storing and managing servers in today’s market. Commercial-off-the-shelf servers (or COTS) usually utilize their own systems within their server modules to make integration as simple as possible, hence the name (Rack Solutions, 2018). Servers today can come in a wide variety of different configurations. One such configuration is the widely popular Server Rack. These come with their own benefits and drawbacks, but are designed to be mounted within the server chassis. These require that each server runs its own set of connections, but can be customized to only fit what you need, with no additional gimmicks to drive up the price. Another popular configuration would be a Blade Server. Unlike the name would suggest, they are not sharp; instead, they are physically thin. This is done deliberately in order to ensure that as many of them fit in as small a space as possible. After all, if your space-constrained as most businesses are, the more servers you can fit in, the more your potential output is going to be. These are designed to be as modular as possible to make swapping out individual blades a simple process. Perhaps one blade is full on its storage, but you would like to keep it for later without using it further. You can simply replace it with a brand new blade. This can be an extremely important factor in high-load environments.

There are many factors to consider when developing logic and ideas for expanding businesses. Technology at the forefront can be extremely high-risk, but also provide a great reward for diligent businesses. It is important to plan each step extremely carefully to ensure that catastrophic failure does not happen, but also to ensure an optimized approach has been implemented and factored upon. This optimization can provide an exponential benefit when business picks up, and clients get more and more demanding.

# **References**

Amaratunga, K. (2000). *Foundations of Software Engineering*. Retrieved from mit.edu: https://ocw.mit.edu/courses/1-124j-foundations-of-software-engineering-fall-2000/pages/lecture-notes/multithreading/

Princeton Research Computing. (2023). *What is a Cluster?* Retrieved from princeton.edu: https://researchcomputing.princeton.edu/faq/what-is-a-cluster

Rack Solutions. (2018, February 13). *Blade Server vs Rack Server*. Retrieved from Rack Solutions: https://www.racksolutions.com/news/data-center-optimization/blade-server-vs-rack-server/

RStudio Connect. (n.d.). *High Availability & Load Balancing*. Retrieved from data-viz.it.wisc.edu: https://data-viz.it.wisc.edu/\_\_docs\_\_/admin/load-balancing/

The University of Rhode Island. (n.d.). *How Computers Work: The CPU and Memory*. Retrieved from uri.edu: https://homepage.cs.uri.edu/faculty/wolfe/book/Readings/Reading04.htm