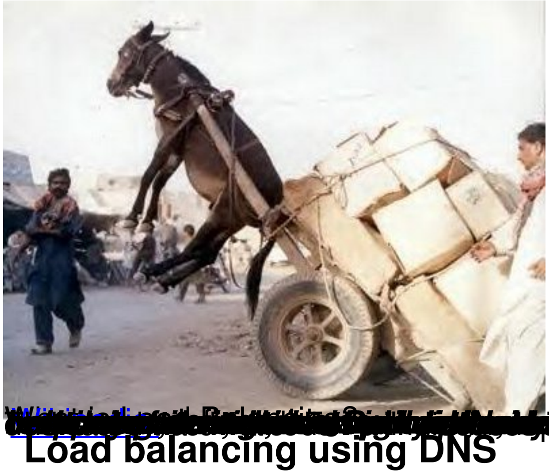
This series is intended to provide documentation of a simple server cluster based on two physical servers and four virtual servers per machine, and configured to host an instance of the Moodle Learning Management System.

Topics

- **Introduction**
- **System hardware and Virtual Server environments**
- **Load Balancing and failover**
- **Web Server, sessions, and updates**
- **File server, repositories, synchronisation and network issues**
- Database implementation, tuning, and replication
- System monitoring and operations
- System trouble shooting

Load Balancing and Failover



[illegible] read this article [illegible]

Load balancing using DNS

The Domain Name System, or DNS as we more commonly refer to it, allows for multiple IP numbers to be assigned to a domain or sub-domain. When a DNS server is served with a request for DNS resolution, it will then deliver a result that sequences through the assigned numbers. We call this **Round Robin** DNS after the childrens game.

This method provdes a very easy, but basic method of load balancing - or rather load distribution as the balancing part is not very evident. There are downsides to this method, due to the various methods of caching inherent in the DNS System itself. The balancing component is very broad, and does not take into account any points of failure. This means that a faulty element in the system will continue to be served requests.

Load balancing using Dedicated Dardware

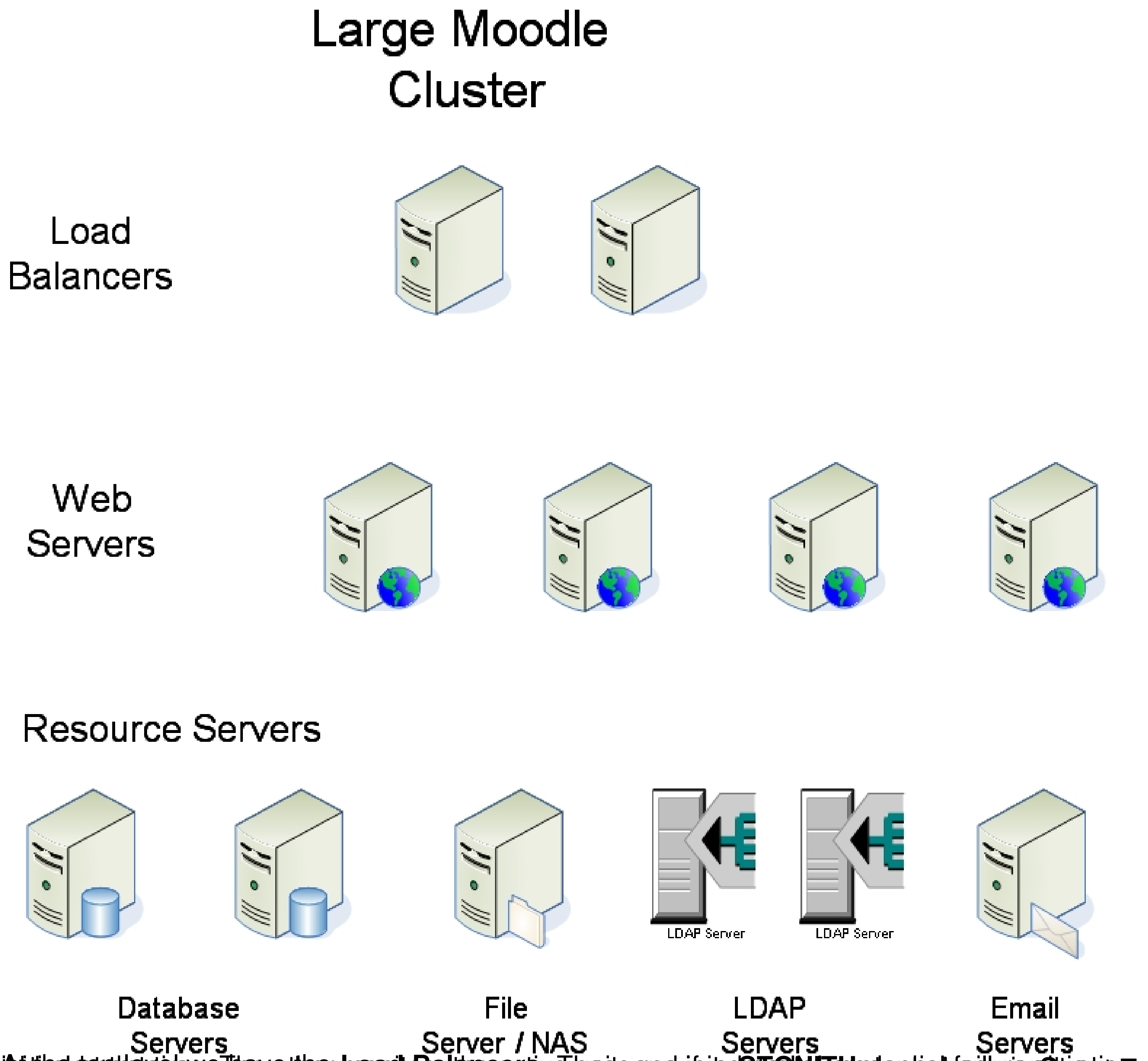
There are proprietary solutions which perform this role. Please refer to the manufacturers documentation for more information in this regard. As a starter try these sites:

5-Load-Balancers-You-Need-to-Know.htm

best-load-balancing-hardware.html

If you want to build your own, then read on. We will be implementing **IPVSADM** with help from the Linux Virtual Server Project. The elements we will be using are **Pacemaker** (wh**ic**h includes Heartbeat) and **ldirectord**. The latter is used in Github for load balancing.

Lets look at a picture of the system we will be implementing:



[illegible]

Pacemaker & Heartbeat

Pacemaker (including Heartbeat) is the process that will provide the high availability of the Load Balancers themselves, so that if one fails, the other will take over almost immediately. As the documentation for installation and configuration changes from time to time I will not repeat/copy it here, but instead refer you to the **source documentation** . However, for installation check out the install link below.

LdirectorD

LdirectorD is the process that manages the actual balancing of the load between the Webservers. here is an excellent article for installing both Pacemaker/Heartbeat and LdirectorD.

Create High-Available Loadbalancer with Pacemaker

With these systems in place you now have Load Balancing and Failover capability for both the Load Balancers themselves, and the Web Servers.

Some other notes

With LdirectorD, you can configure a default message should all the Web Servers become unavailable.

Next

In the next article we will look at the setup for the webservers, handling sessions, and the update management of the Moodle application code.