C R E S C E N D O N E T W O R K S

Server Load Balancing With The Maestro Application Delivery Platform

April 2005

CN_WPT_240_0405

Telephone Contact:

International
Tel. +972.3.634.6120

In the US
Tel. +1.866.830.0400

Web Address:

www.crescendonetworks.com

Email:
CNInfo@crescendonetworks.com

Orchestrate Your Business

Crescendo Networks Ltd.

www.crescendonetworks.com

Crescendo Network's Maestro Application Delivery Platform

Crescendo Networks provides high-performance application delivery and acceleration for enterprises and web sites. The unique design of Crescendo's Maestro Application Delivery Platform brings instant performance relief to overburdened data centers today - and Maestro is the only solution built to be an integral component in emerging data center architectures.

Crescendo's Maestro Application Delivery Platform is a multi-gigabit, hardware-based platform that is capable of offloading task-intensive functions from servers in a web application, optimizing that application and allowing the servers that host it to scale significantly. The Maestro Platform is an appliance that front-ends the servers, intercepting and processing all user requests destined for them. By performing this functionality, Maestro can provide various optimization services for the servers in the application, massively improving server performance while reducing user response time and consumed bandwidth.

The Need for Load Balancing

Today's web applications are consistently deployed in multi-server networking environments. This is done for two primary reasons: scalability and fault tolerance. Having multiple servers allows an application to grow with user demand while protecting itself from the failure of any single element. However, users still need only a single target address (e.g. URL or IP address) for an application, which is a complication when the application is made up of multiple physical machines.

The earliest, and most primitive, way to solve this problem was to use DNS configuration to somewhat methodically direct users to a different server. However, this solution not only distributed users across the servers unevenly, it was unable to provide quick failover in case one of the servers failed. This resulted in the emergence of server load balancing as a technology, either deployed as a standalone appliance in front of the server farm or as a software module on the servers themselves. Hardware load balancers are more robust since they are dedicated devices and don't spend CPU resources on the server for load balancing tasks.

As a technology, load balancing has come a long way. What used to be simply directing TCP connections to servers has evolved into logic that can make decisions based on Layer 7 information, account for client persistency, employ advanced algorithms for picking a server, and recognize server failure at the application layers.

Load Balancing with the Maestro Platform

Crescendo's Maestro Platform is a natural point in the network for deploying server load balancing logic. Maestro front-ends servers and has full visibility into the request/response chain, thus controlling the delivery of all user requests and the subsequent server responses. Since it maintains optimized TCP connections with each of the servers, load balancing logic is a natural extension of its capabilities. Actually, the fact that Maestro inherently operates at the HTTP level, makes it a better candidate for load balancing than a switch or router that intrinsically operates at Layers 3 or 4.

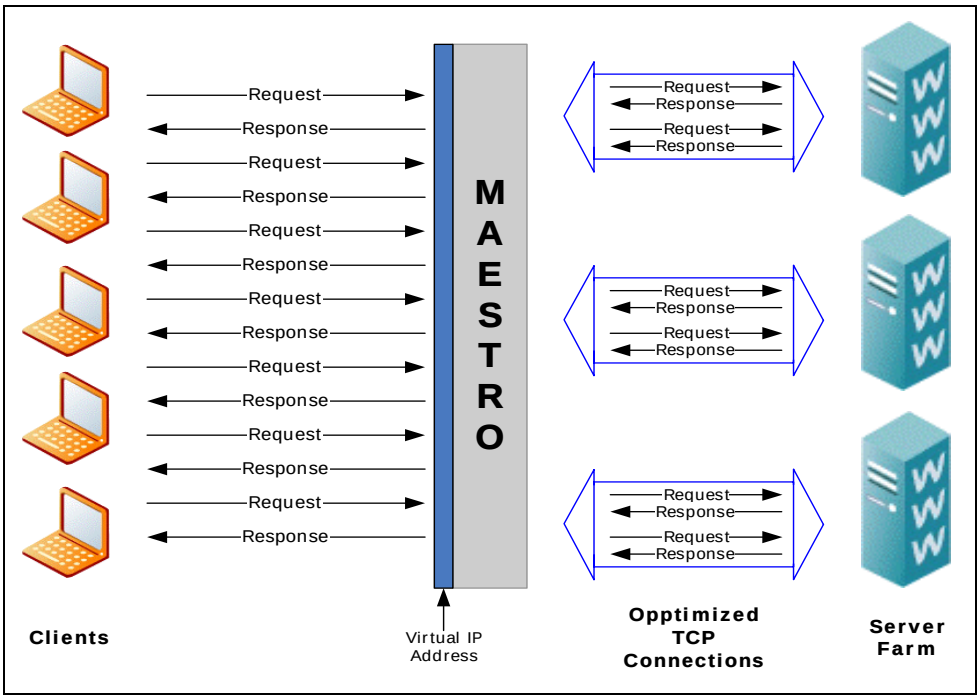
Much like all its other functionality, the breadth of load balancing features on Maestro is deployed over its hardware-based architecture. All load balancing functions are performed by hardware components that contribute to the scalability and the robustness of the platform.

Beyond Simple Load Balancing

Because of its inherent Layer 7 functionality, Maestro's load balancing feature set goes beyond simply directing client requests to an available server. Maestro's load balancing operation can perform both simple and advanced functionality needed for today's web applications.

Identical servers are configured together in *Clusters,* each individually managed by Maestro. This means that Maestro holds optimized TCP connections with each server, always aware of the number of requests that the server is handling at any given moment. At the same time, the server is also actively and periodically polled for health, assuring that requests never go to a server that is incapable of handling them.

Virtual Servers are then configured in IP address and TCP port pairs. Each Virtual Server is then mapped to one ore more clusters according to a number of optional Layer 7 rules. The rules can select a cluster according to file type, URL path, hostname, or language. Multiple criteria can also be configured within a single rule, providing extreme flexibility for cluster selection. Once a cluster is selected, Maestro directs the client request to the most appropriate server. Although simple round robin distribution of requests is available, Maestro's inherent ability to know the number of pending requests for each server gives it an additional powerful algorithm for server selection. By choosing the server with the least number of pending requests, Maestro can select the most appropriate server based on actual request load on each server; something that standard load balancers are not capable of doing. With both algorithms, stronger servers can be configured to receive more of the client requests.



Server Load Balancing Through Maestro

Many web applications also have specific requirements regarding client persistence – the ability of a user to stay connected to the same server for the duration of its session. Maestro's load balancing feature set supports both applications that require client persistence and those that don't. Persistency can be maintained either by IP address or by HTTP cookie. This ensures that persistence can be maintained even in applications where a client IP address is not an adequate enough identifier for a specific user. For applications that do not require persistence, requests are distributed between servers based on the conditions in the network only at the time the request arrives.

Integrated Functionality

Maestro's comprehensive load balancing feature set is fully integrated with all of its application optimization and server offload functionality. Load balancing can be used together with SSL offload or content compression, for example, while still being able to use all the TCP optimization functionality of the device, such as connection consolidation and request/response buffering. Load balancing rules can also be applied to secure requests, whether they're sent to the server via plain HTTP or SSL. Additionally, because all services are handled in dedicated, task-specific hardware modules, multiple services can be enabled at once with no effect on Maestro's performance or functionality.

As more functions become available to the Maestro Platform, this level of integration will continue, allowing all services to be used concurrently and seamlessly in a fully integrated manner. At the same time, the powerful underlying architecture of the device will allow all functions to operate together without any degradation to the performance of the device.

Conclusion

The Maestro Platform provides a comprehensive and flexible load balancing feature set that allows it to efficiently distribute user requests across clusters of identical servers. Additionally, since Maestro is in control of the actual request flow to the servers, it can direct traffic to them based on real request load; a metric not available to standard load balancers.

All load balancing functionality is fully and seamlessly integrated with all other optimization services provided by the highly scalable, multi-gigabit Maestro Platform. At the same time, because of its unique and powerful task-specific, hardware-based architecture, all services can operate concurrently without any degradation in device performance.