



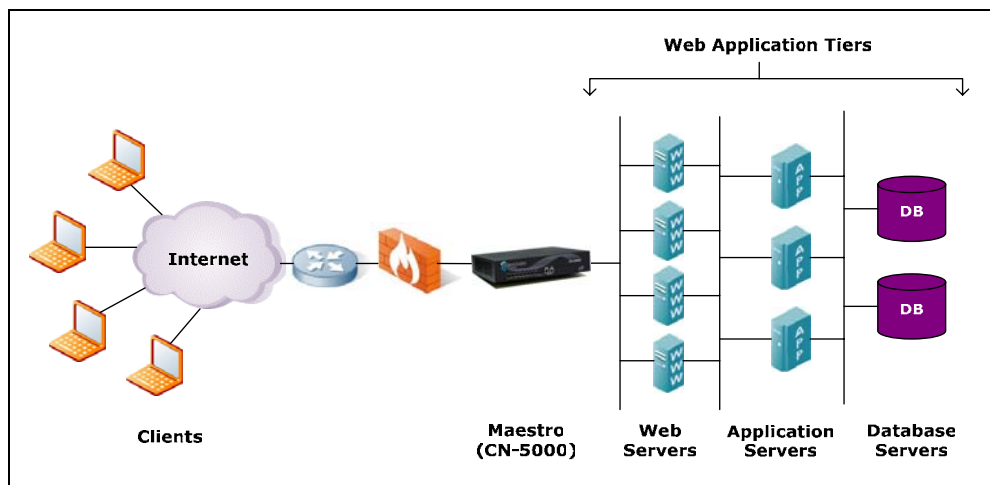
An Introduction to Crescendo's Maestro Application Delivery Platform

Introduction

This document is intended to serve as a short introduction to Crescendo's Maestro Platform and its core features/benefits. The document will also highlight competitive advantages of the product, along with typical customers best suited for the platform.

Functionality

Maestro is a purely hardware-based appliance that front-ends web applications in order to optimize/accelerate the application. This class of product is commonly referred to as an AFE (Application Front End). Maestro (specifically, the CN-5000 Maestro product family) is functionally located in a network as seen in the diagram below:



Typical Maestro Deployment

At this location in the network, Maestro performs a number of optimization/acceleration tasks:

TCP termination, offload, and connection consolidation:

Description: Maestro is the actual TCP endpoint for all client connections. In turn, it starts and owns a small number of server side connections. All incoming requests are received by Maestro over the client-side connections and then directed to one of the server side connections. This not only helps reduce the TCP overhead for each server, but also accelerates the rate at which each server delivers content to the network.

Benefits: When it comes to handling TCP connections, standard server TCP stacks have 3 major pain points:

1. Handling a large number of TCP connection setups and teardowns
2. Handling a large number of simultaneous TCP connections
3. Dealing with WAN-based connections that are subject to delays, drops, retransmission, and congestion

Web applications facing Internet clients expose servers to all of these pain points. Maestro addresses all these issues by performing TCP termination, offload, and connections consolidation. The connections between the CN-5000 and the servers are small in number and long in lifetime. This relieves the server from both the number of setup/teardown operations it must perform and the number of connections it must simultaneously maintain. Also, since Maestro initiates and owns the server-side connections, the servers essentially think all the clients are on the same LAN. This allows them to be totally shielded from WAN-imposed connection issues and transmit data to the network at full LAN speeds. Maestro takes care of all buffering necessary to transmit the responses to the clients, based on the individual capabilities/bandwidth of each client connection

Content Compression:

Description: Since Maestro is fully responsible for delivering the server responses to the clients, it can perform HTTP content compression as the request flows from the server to the client. This is done through a fully hardware-based compression module that can compress at 1Gbps wire speed.

Benefits: Content compression reduces the number of bytes that flow from the web site to the client. This helps reduce the amount of bandwidth used by the site, which can help any site that is limited in bandwidth usage, either technically (only finite amount of bandwidth available) or financially (higher cost for more bandwidth utilization). Content compression also reduces the number of bytes that the client needs to receive for each object, effectively improving the client's perceived response time.

Compression is also an offload technology. The benefits of compression are vast enough that servers may, themselves, perform the content compression. However, compressing content is a CPU burden for the server which increases the overhead for content serving. Offloading the compression task to the CN-5000 relieves the server from this pain.

SSL Offload:

Description: Maestro can also offload all SSL processing from the server. All SSL connections are terminated at the CN-5000 and processed at scale through an embedded hardware offload module. The SSL module can handle both SSL session setup and bulk data encryption tasks. Secure requests can be sent to the servers via clear-text HTTP or via SSL (using weaker security ciphers and longer lasting SSL sessions, for minimized impact). The server response is also re-encrypted by Maestro on the way back to the client.

Benefits: SSL processing is a significant resource burden for a server. The cryptographic algorithms that are used in both session setup (SSL handshake)

and actual data transfer (bulk encryption) both require significant CPU cycles from a server. Maestro offloads this functionality to dedicated hardware that accelerates both session setup and bulk encryption. The advantage of performing the offload external to the server is that additional functions such as compression and load balancing can be performed on secure data since Maestro has access to the clear-text version during the encryption/decryption process.

Load Balancing:

Description: Load balancing allows Maestro to distribute the incoming requests across a number of identical servers. Because Maestro knows the specific status of each request with each server, it can optimally choose the most appropriate server to fulfill each request. Requests can also be directed to different clusters based on URL, file type, language, or Host header.

Benefits: Load balancing not only allows a site to scale with new servers, but also provides failover and fault tolerance in case servers in a web application have unexpected failures. It also provides a seamless way for taking servers in and out of service for maintenance reasons. Maestro's load balancing functionality is request-based, because of the underlying architecture and TCP termination capabilities. Maestro can also be integrated into a network with an existing load balancer, allowing it to add its offload and acceleration functionality to the network.

Other Functionality:

The following are a few of the other highlights of the maestro platform:

- DDoS Protection – Since the maestro platform is built from the ground up to deal with HTTP traffic and WAN clients, it's innately immune to common L2-L4 DDoS exploits, especially SYN attacks.
- Fast TCP – Maestro's hardware-enabled TCP engine incorporates optimized TCP algorithms that provide two improvements over standard TCP implementations:
 - A mechanism to speed up TCP's slow-start algorithm, in order to reach maximum capacity of a TCP link quicker.
 - An optimized congestion avoidance algorithm that not only tries to use the maximum allowable bandwidth of a TCP connection, but minimizes dropped packets.

Competitive Advantages

Maestro has three significant advantages over other AFE solutions which provide similar functionality:

Device Architecture

The Maestro is purely hardware-based system based solely on hardware components such as network processors and FPGAs. The system has dedicated hardware for each of its many tasks that utilize more than 60 micro-engines for a truly distributed architecture. Similar products are built on off-the-shelf PC platforms that utilize common operating systems and a

central processor model. Maestro's architecture gives it significant advantages that translate to better performance and scale.

Scale

Because of its architecture, Maestro's platform can achieve significant scale and has performance benefits over the competition. The platform is capable of handling more than 3.6Gbps sustained web traffic. Furthermore, the purely hardware-based compression engine can perform compression at 1Gbps wire speed. The competitive software-based products cannot scale at these levels.

Feature Concurrency

Since all tasks are handled by dedicated, isolated hardware, Maestro is capable of providing multi-functional capabilities without any degradation in performance. In other words, not only are all features integrated together seamlessly, but they can all be enabled with no one feature affecting the performance of another. All functions can operate concurrently at scale. The competitive PC-based products which are based on a central processor model are designed so that all functions use the same processing pool. Because of this, when multiple functions are enabled, the overall device performance suffers often causing it to become a network bottleneck.

Typical Applications

Maestro's offload, acceleration, and optimization functions can be applied to any web application using the HTTP protocol. Maestro's inherent HTTP intelligence allows it to help any web server or application. However, the platform becomes more applicable to a web environment if any or a combination of the following issues is a concern:

- Bandwidth over-usage – Maestro's compression capabilities can significantly reduce the amount of outbound bandwidth used by a web application.
- User response time improvement – A number of Maestro functions can help improve the perceived response time experienced by the clients of an application:
 - Compression reduces the number of bytes to be received by a client, directly improving the user experience.
 - TCP offload relieves the server from spending its resources on TCP connection handling. Since Maestro is inherently better in dealing with client connections than a server, the client response times improve when the user has TCP connectivity to Maestro instead of the server.
- Server resource utilization – In situations where server CPU cycles (and memory usage) are precious, Maestro can dramatically reduce the resource utilization of a server, allowing it to allocate all of its resources to serving content, rather than the related overhead. The following functions help relieve the server:
 - TCP offload significantly reduces the amount of resources a server allocates to dealing with TCP connections.
 - Compression offloads the task from the server, freeing up the CPU from dealing with this task. The application, however, can still enjoy the benefits of content compression since Maestro provides this function external to the server.

- SSL offload provides a much-needed relief for the servers, freeing them from dealing with the task-intensive cryptographic algorithms that today's secure applications demand.
- Peak load protection – Maestro's functionality inherently allows a server to scale and handle much more traffic than it could without being front-ended by Maestro. Typically, a server can handle more than 3-5 times the amount of traffic after Maestro deployment. Because of this, a server infrastructure can easily handle peak traffic conditions, since typical server utilization will be drastically reduced after Maestro is deployed.
- High availability and fault tolerance – Maestro's load balancing functionality allows many servers to host a single application. This not only allows the application to scale by having new servers added, as necessary; but also provides fault tolerance and high availability for the servers.

Anticipated Results

The following table outlines expected improvements in server resource utilization, site bandwidth usage, and end-user response time:

TCP Offload	
Server CPU Util. Savings:	Up to 50%
Server Mem. Usage Savings:	Up to 80%
Server Capacity Improvement	More than 300%
Compression	
Client Response time Improvement:	Up to 66%
Site Bandwidth Savings	Up to 66%
Server CPU Util. Savings:	Up to 90%
SSL	
Server CPU Util. Savings:	Up to 90%

*All tests performed with CN-5000 and Apache 2 server

Conclusion

Crescendo's Maestro Application Delivery Platform is a significant enabler for web applications by providing essential functions while offloading tremendous overhead from servers. By allowing the servers to spend their resources on the application itself, rather than its overhead, the servers scale better and can reach their true potential. Meanwhile, functionality such as compression and load balancing allow the application itself to operate more effectively, use less bandwidth, and have optimal response times for its users.