April 2003

ColdFusion MX for IBM WebSphere Application Server Performance Brief

The information contained in this document represents the current view of Macromedia on the issues discussed as of the date of publication. Because Macromedia must respond to changing market conditions, it should not be interpreted to be a commitment on the part of Macromedia, and Macromedia cannot guarantee the accuracy of any information presented after the date of publication.

This Guide is for informational purposes only. MACROMEDIA MAKES NO WARRANTIES, EXPRESS OR IMPLIED, IN THIS DOCUMENT.

Macromedia, Inc • 275 Grove Street • Newton • MA • 02466

www.macromedia.com • info@macromedia.com • (617) 219-2000

© 2003 Macromedia Inc. All rights reserved.

Table of Contents

Executive Summary

ColdFusion MX for IBM WebSphere Application Server delivers greater application performance than any prior version of ColdFusion. This product has been designed and tuned for maximum performance and scalability by leveraging the power of the underlying J2EE server. It takes full advantage of the performance optimizations available with the IBM Java virtual machine (JVM) version 1.3.1 and provides its own performance-enhancing features, including query and page caching, support for in-memory queries, and built-in facilities for profiling code and identifying bottlenecks.

This brief provides detailed performance and scalability information about ColdFusion MX for IBM WebSphere. Running on the enterprise-class WebSphere Advanced Edition 4.0.3, ColdFusion MX also delivers exceptional performance, scalability and reliability.

Statistics Summary

- Windows – As much as 2.5 times faster than ColdFusion 5
- Solaris – More than 2 times faster than ColdFusion 5
- Linux – 3.5 times faster than ColdFusion 5

Application Performance

Defining Application Performance

A high performing application is able to deliver content to users quickly. In this brief, application performance is analyzed by measuring page response time: the elapsed time between the submission of a request (the clicking of a submit button, the manual entry of a URL, or the clicking of a link by a user) and the successful completion of that request. A lower response time per request (completing the request more quickly) allows the application to deliver more content overall to more users in a given period of time (throughput).

There are three basic factors that influence response time performance:

- Web application server architecture and configuration
- Network infrastructure
- Web page design

All three factors are of considerable importance, and no Web application will function efficiently if any factor is neglected. However, since network bandwidth and Web page design are not functions of the application server, this brief focuses solely on the changes in page response time that result from enhancements to the ColdFusion Server.

Performance Improvements in ColdFusion for IBM WebSphere

One of the major changes made in the ColdFusion MX release affects the execution of ColdFusion templates. In previous release, ColdFusion pages were processed by an interpreter with every request. In ColdFusion MX, this has been changed. When a page is first requested of ColdFusion MX Server or ColdFusion MX for IBM WebSphereJ2EE Application Servers, the page is compiled into Java bytecode, at which point it is executed and then cached to memory and disk. All subsequent requests for that page are handled by the cached bytecode, unless an update to the page forces the page to be recompiled. As this brief illustrates, this change results in a significant gain in performance for production applications. Other factors that affect performance are the ability to take advantage of the high-performance features in the application server on which ColdFusion MX is deploy, such as database connection pooling, process cloning, and clustering.

However, because the first call to a ColdFusion page causes compilation, some users may observe causes reduced performance on initial viewing of an application. On the next and subsequent requests, however, response times are much faster (see Illustration 1).

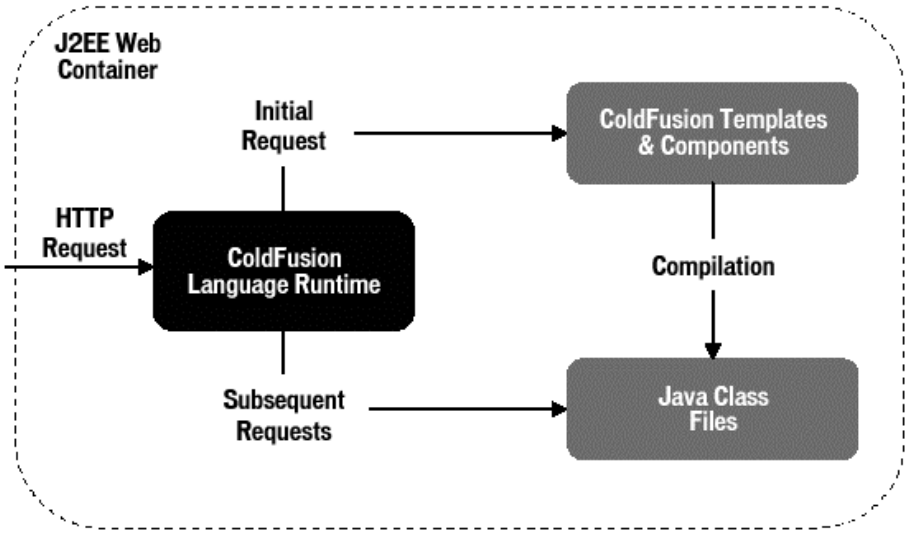


Illustration 1: ColdFusion MX for IBM WebSphere Processing

Test Configuration and Application

Testing for this brief was performed with ColdFusion 5 Enterprise Edition, ColdFusion MX Server Enterprise Edition, and ColdFusion MX for IBM WebSphere on single servers with varying multi-processor configurations on Microsoft Windows 2000 Advanced Server with IIS 5.0, Sun Solaris with Netscape Enterprise Server 4.0 for CF5 and CFMX, and IBM HTTP Server for WebSphere, and Red Hat Linux with Apache HTTP Server version 1.3.22 for CF5, Apache 2.0.40 for CFMX, and IBM HTTP Server for WebSphere. ColdFusion MX for IBM WebSphere testing was performed with IBM WebSphere Advanced Edition 4.0.3 Analysis of the results is presented in following sections grouped by operating system.

For each operating system, data is presented comparing the performance of the three ColdFusion product versions. On Windows 2000, additional data is provided illustrating the ability of ColdFusion MX for IBM WebSphere to provide linear SMP scalability as the number of processors increases resulting in higher throughput.

Settings were adjusted to achieve optimal performance in each product version. In ColdFusion 5 Enterprise Edition and ColdFusion MX Enterprise Edition, adjustments were made to the server settings for simultaneous requests. This setting is managed by the underlying application server and indicates the number of simultaneous requests that ColdFusion will serve at any given time. When the server reaches the specified limit, requests are queued and handled in the order received. The general rule is to set the simultaneous requests setting to 3-4 times the number of CPUs on the system. So, if the server has two processors, the number of simultaneous requests for the ColdFusion server should be set to 6-8. In this case, if and when there are more than six threads being requested, ColdFusion automatically queues additional requests until one of the other two is freed up. Note that the optimal number may vary depending on hardware configuration and the application being run. Real-world deployments of ColdFusion applications will benefit from the configuration of each application's particular optimal simultaneous request setting.

In ColdFusion MX for IBM WebSphere there is no such simultaneous request setting. In that version's testing, adjustments were made to the minimum and maximum thread size settings for the Web Container Service with a high thread inactivity timeout setting (see the Appendix for the settings used).

ColdFusion MX Server Enterprise Edition and ColdFusion MX for IBM WebSphere tests were also conducted with the ColdFusion "trusted cache" setting enabled. This specifies that any requested files found to currently reside in the template cache will not be inspected for potential updates. For sites where templates are not updated during the life of the server, this minimizes file system overhead. This is an important configuration setting to ColdFusion MX performance in general and should be used appropriately in all application deployments.

Testing for this brief was performed in Macromedia's Enterprise Testing facility in Newton Massachusetts. The application used in the test is the Tack2++ sample application, a new version of the original Tack2Plus sample application included with ColdFusion MX. It preserves all of the functional characteristics of the original Tack2Plus application but makes use of template and query caching features, as would a real-world production ColdFusion Web application. The application represents a typical e-commerce Web application found on the Web. For more details on the application, see the Appendix.

Test Results

Windows 2000

Two different server configurations were used for Windows 2000 testing.

SMP Scalability Testing:

- Windows 2000 Server SP2
- Intel OCPRF100 Server
- 1, 2,4, and 8 x 500MHz PIII Xeon processors
- 4 GB RAM

Performance Comparison Testing:

- ColdFusion Server 5 Enterprise Edition
- ColdFusion MX Server Enterprise Edition
- ColdFusion MX for IBM WebSphere
- IBM WebSphere Application Server Advanced Edition 4.0.3
- Windows 2000 Server SP2
- Compaq 1850 Server
- 2 x 500MHz PIII Xeon processors
- 512 MB RAM

The back-end database server for each test suite was a separate Compaq 6500 with four 500MHz Pentium III Processors running Microsoft SQL Server 2000. All tests were performed using 100 virtual users (VUs) simulated with Segue SilkPerformer V.

To demonstrate linear SMP scalability with ColdFusion MX for IBM WebSphere, the processor configuration of the test machine was gradually increased from one to eight processors. As illustrated in Figure 1, the average page response time for the application improved in a near-linear fashion as processors were added. The results also showed that ColdFusion MX for IBM WebSphere provides superior response times over other versions of the product.

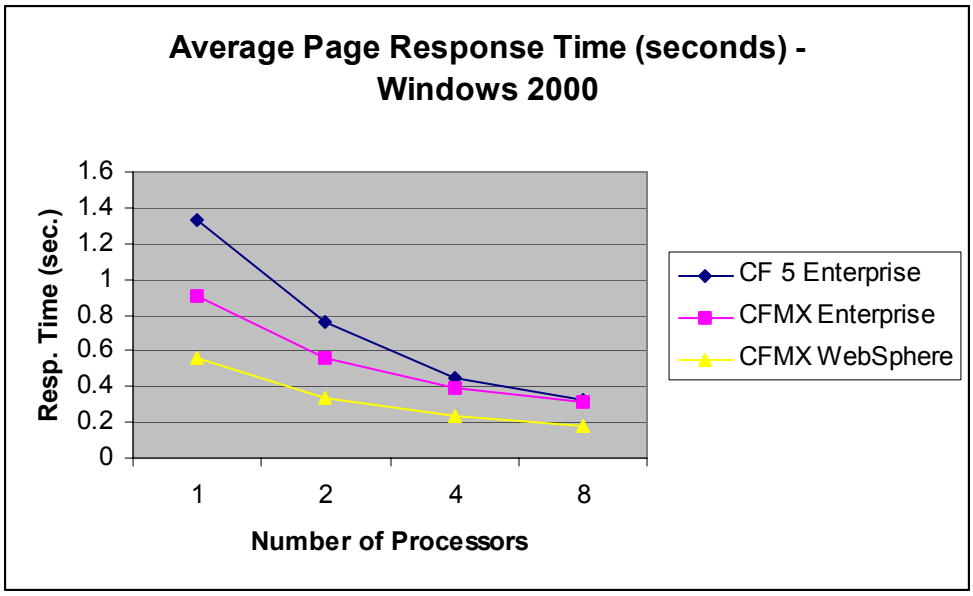


Figure 1: Near-Linear improvement in average page response time with addition of processors on Windows 2000 and up to 2.5 times faster than earlier versions of ColdFusion.

In addition to processing page requests more quickly, ColdFusion MX for IBM WebSphere scales to a much greater degree than other ColdFusion product versions as the number of server processors is increased. This results in the application being able to service more requests in a given period of time (as illustrated in Figure 2).

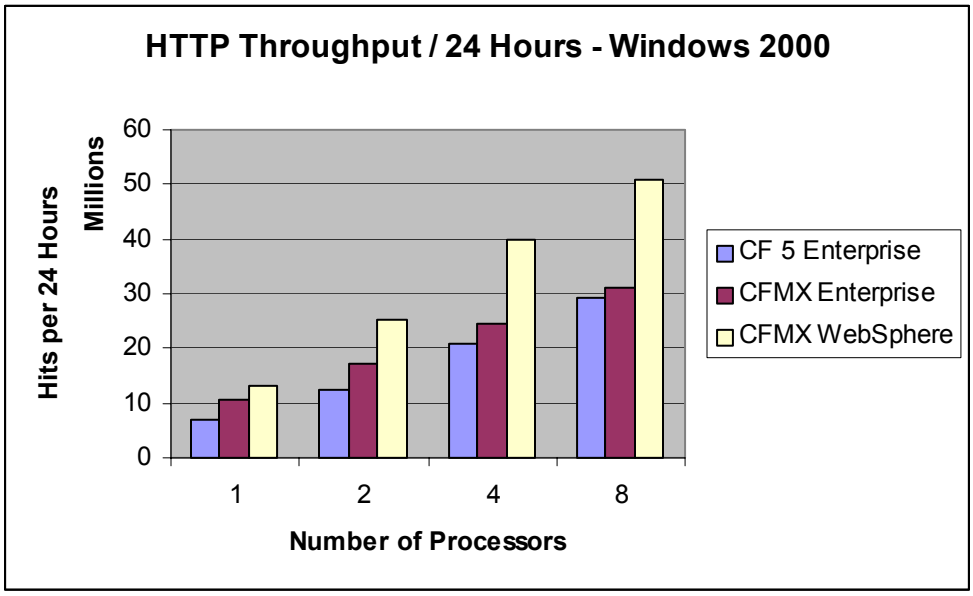


Figure 2: HTTP Requests served per 24 hours on Windows 2000 across three product versions and multiple processor configurations.

Solaris

The performance comparison between ColdFusion 5 Enterprise Edition, ColdFusion MX Enterprise Edition and ColdFusion MX for IBM WebSphere on Solaris was conducted on a server with the following specifications:

- IBM WebSphere Application Server Advanced Edition 4.0.3
- Solaris 7
- Sun E220 Server
- 2 x 450MHz SparcV9 processors
- 512 MB RAM
- IBM HTTP Server (Apache)

The JVM for Solaris is Java(TM) 2 Runtime Environment, Standard Edition (build JPSE_1.3.1_20020313) with the Java HotSpot(TM) Client VM (build JPSE_1.3.1_20020313, mixed mode).

The back-end database for each test suite was a separate Compaq 6500 with four 500MHz Pentium III Processors running Microsoft SQL Server 2000. All tests were performed using 100 virtual users (VUs) simulated with Segue SilkPerformer V.

As was observed with Windows 2000, ColdFusion MX for WebSphere demonstrated considerable performance gains over previous versions of the product. The average page response time was less than half that of ColdFusion 5, as illustrated in Figure 3.

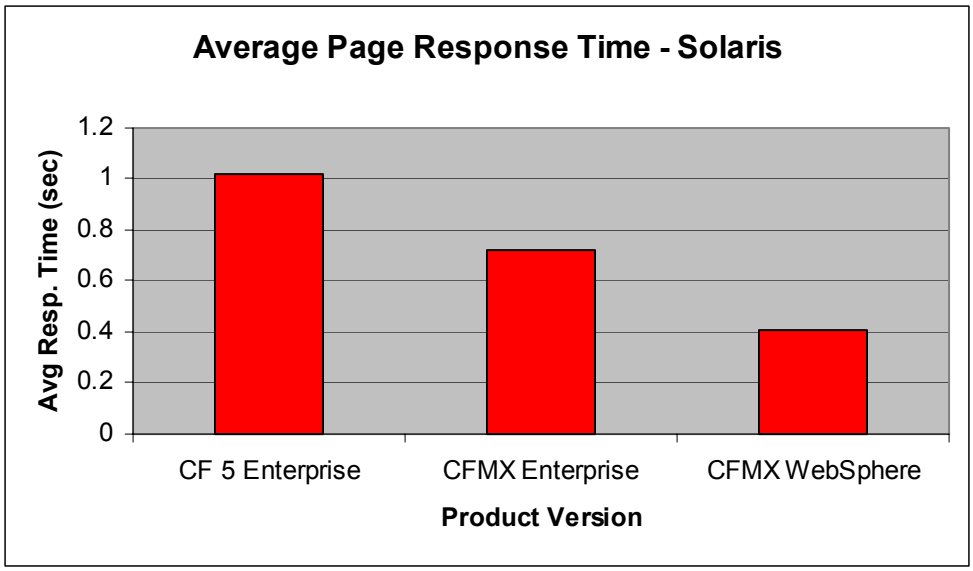


Figure 3: Improvements in average page response time with three product versions on Solaris.

In addition to faster delivery of content, scalability improvements in ColdFusion MX for WebSphere result in the ability to deliver more content than other versions of ColdFusion over a given period of

time. Figure 4 shows that more than two and a half time as much data was served in a 24-hour period.

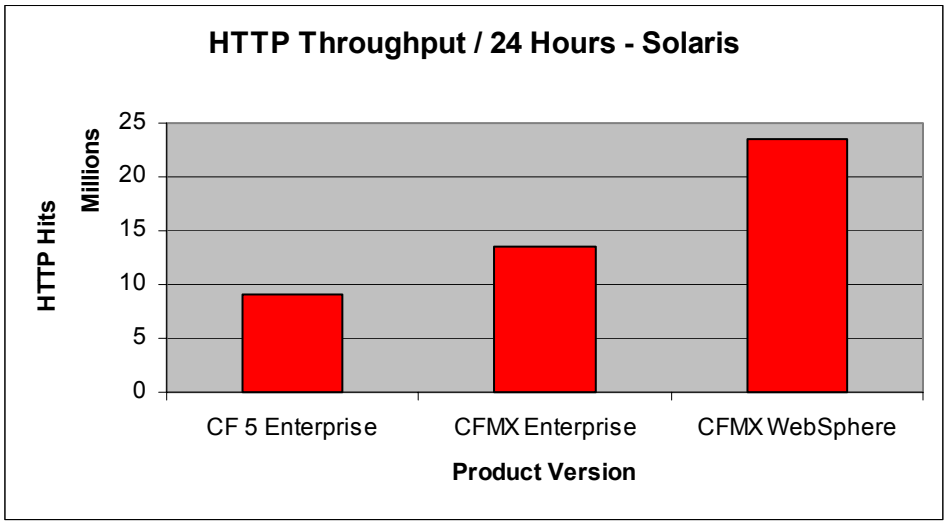


Figure 4: HTTP requests served over a 24-hour period on Solaris.

Linux

The performance comparison between ColdFusion 5 Enterprise Edition, ColdFusion MX Enterprise Edition and ColdFusion MX for IBM WebSphere on Linux was conducted on a server with the following specifications:

- IBM WebSphere Application Server Advanced Edition 4.0.3
- RedHat 7.2
- Penguin Computing server
- 2 x 933MHz PIII Processors
- 512 MB RAM
- IBM HTTP Server (Apache)

The back-end database for each test suite was a separate Compaq 6500 with four 500MHz Pentium III Processors running Microsoft SQL Server 2000. All tests were performed using 100 virtual users (VUs) simulated with Segue SilkPerformer.

Like Windows and Solaris, Linux testing demonstrated ColdFusion MX for IBM WebSphere to deliver more content with a faster page response time than previous ColdFusion versions.

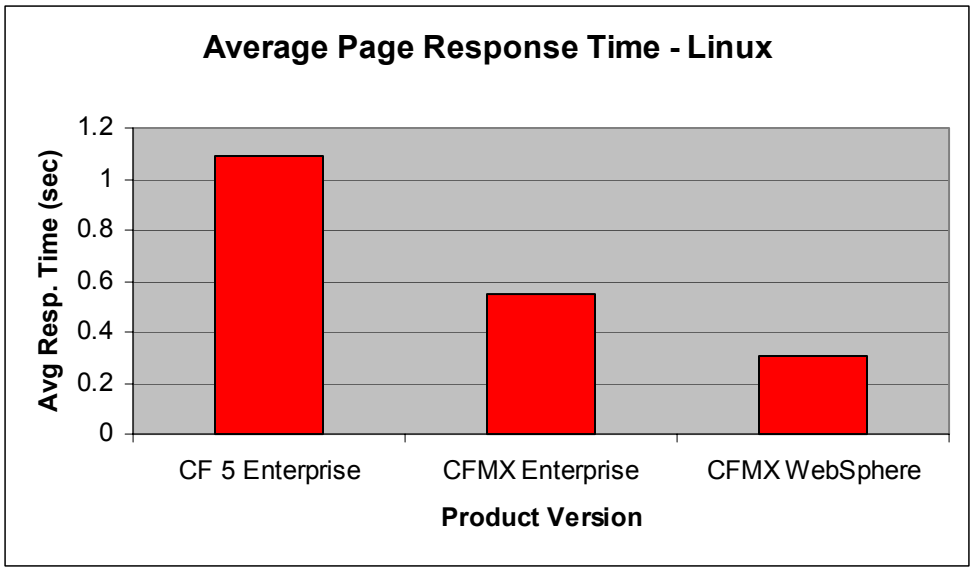


Figure 5: Improvements in average page response time with three product versions on Linux.

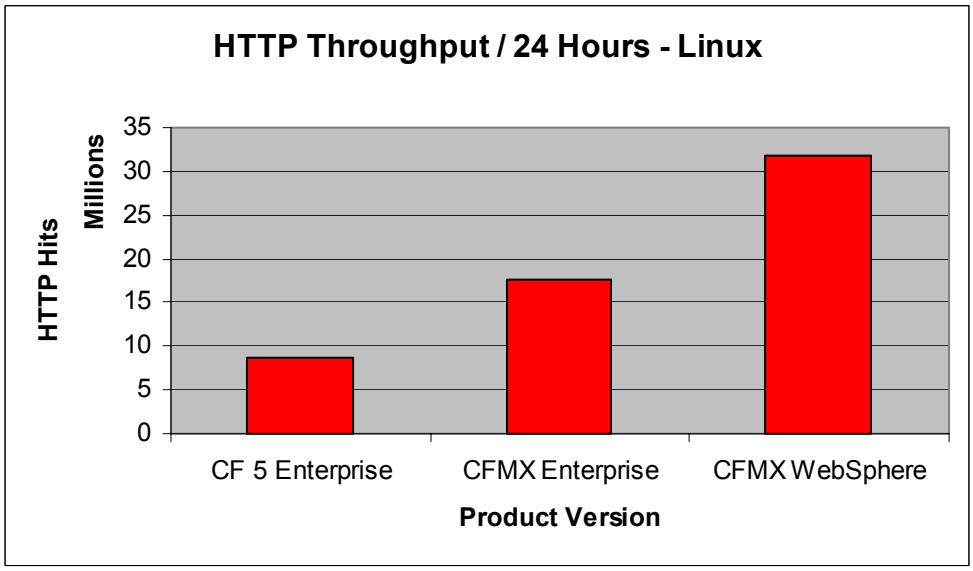


Figure 6: HTTP requests served over a 24-hour period on Linux.

AIX

Macromedia ColdFusion MX for IBM WebSphere is the first version of ColdFusion to run on the AIX 5L UNIX operating system. As such, a comparison between its performance and earlier versions of ColdFusion is not possible. Instead, Macromedia conducted a comparison between AIX and Microsoft Windows 2000 as a deployment platform for Macromedia ColdFusion MX for IBM WebSphere on WebSphere Application Server 4.0.4 Advanced Edition.

Please note that both hardware and operating systems in this test are quite different. Any comparison between these systems should take into account factors in addition to performance; the relative cost of purchasing the systems and skills available among administrators, for example.

The AIX test machine configuration included:

- AIX 5L v5.1 operating system
- 4 x 450 Mhz Power 3-II processors
- 4 GB RAM

(complete machine specifications at http://www-132.ibm.com/content/home/store_IBMPublicUSA/en_US/eServer/pSeries/entry/44P270.html)

The Microsoft Windows server had the following configuration:

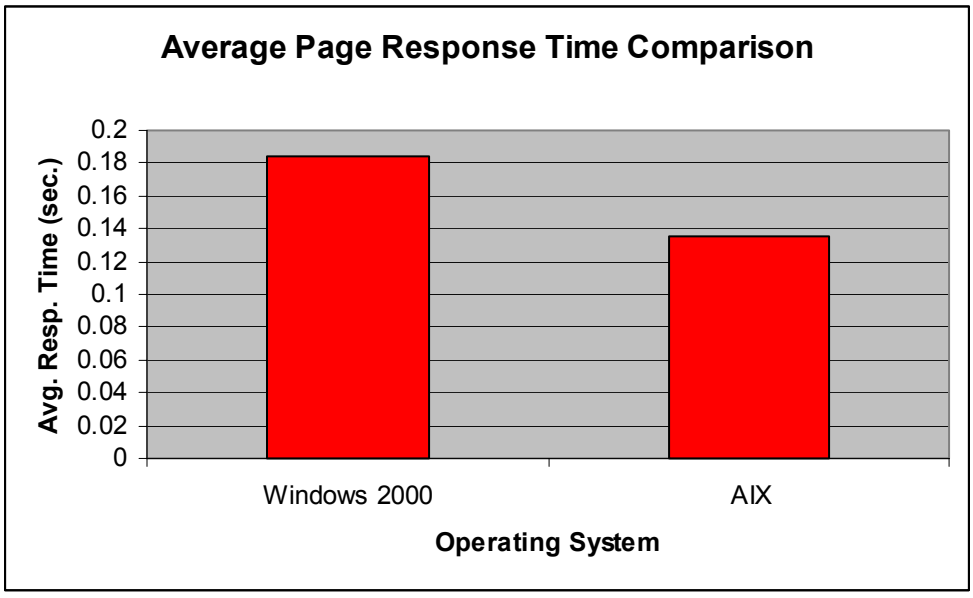
- Windows 2000 Server SP2
- Intel OCPRF100 Server
- 8 x 500MHz PIII Xeon processors
- 4 GB RAM

Both Windows and AIX used WebSphere Application Server 4.0.4 Advanced Edition.

Improvement in Average Page Response Time

On AIX, an average 36% increase over Windows 2000 was measured.

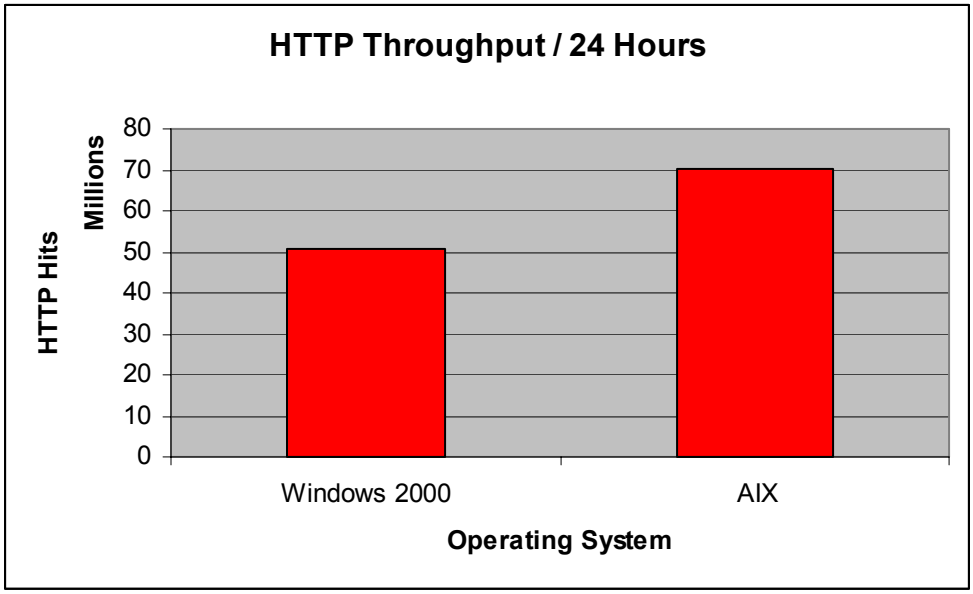
Average Page Response Time (seconds)	
Windows 2000	AIX
0.184	0.135



Improvement in HTTP Throughput

AIX demonstrated the ability to deliver nearly 20 million more HTTP hits in a 24-hour period compared to Windows 2000.

HTTP Throughput / 24 Hours	
Windows 2000	AIX
50,931,072	70,046,208



Appendix

Test Results

Windows 2000 Response Time (seconds)

Product Version	Number of Processors			
	1	2	4	8
ColdFusion Server 5 Enterprise	1.337	0.763	0.452	0.325
ColdFusion MX Server Enterprise	0.911	0.558	0.395	0.318
ColdFusion MX for IBM WebSphere	0.555	0.334	0.231	0.184

Windows 2000 HTTP Throughput (24 hours)

Product Version	Number of Processors			
	1	2	4	8
ColdFusion Server 5 Enterprise	7,060,608	12,332,736	20,940,768	29,108,160
ColdFusion MX Server Enterprise	10,456,128	17,150,400	24,491,808	30,938,976
ColdFusion MX for IBM WebSphere	13,298,688	25,306,560	39,773,376	50,931,072

Solaris Response Time

Product Version	Response. Time (sec.)
ColdFusion Server 5 Enterprise	1.016
ColdFusion MX Server Enterprise	0.72
ColdFusion MX for IBM WebSphere	0.409

Solaris HTTP Throughput (24 hours)

Product Version	HTTP Hits
ColdFusion Server 5 Enterprise	9,144,576
ColdFusion MX Server Enterprise	13,536,288
ColdFusion MX for IBM WebSphere	23,602,752

Linux Response Time

Product Version	Response Time (sec.)
ColdFusion Server 5 Enterprise	1.091
ColdFusion MX Server Enterprise	0.549
ColdFusion MX for IBM WebSphere	0.307

Linux HTTP Throughput

Product Version	HTTP Hits
ColdFusion Server 5 Enterprise	8,671,104
ColdFusion MX Server Enterprise	17,616,960
ColdFusion MX for IBM WebSphere	31,816,800

WebSphere Web Container Service Settings

The following Web Container Service settings were found to be optimal for this ColdFusion MX for IBM WebSphere testing.

# of Processors	Thread Size (Min/Max)	Thread Inactivity Timeout (seconds)
1	3/3	999
2	8/8	999
4	12/12	999
8	12/12	999

Tack2++ Test Application Details

The Tack2++ application represents a typical e-commerce application consisting of a catalog, shopping cart, check out pages, etc. While actual performance gains experienced with other applications will vary depending on scope and complexity, customers can expect to realize similar results with similar applications and server configurations.

The user load mix used in this round of testing is generally more demanding than is typically experienced on public e-commerce sites. In the test suite, each virtual user initiates and completes a purchase, whereas studies have shown that approximately 90% of all traffic on a typical e-commerce site consists of browsing activities, and less than 5% of all users actually add items to the shopping cart and check out.

Each user session in the test suite consisted of the following series of requests:

1. Main Menu
2. Show Items
3. Show Specific Item
4. Add Item to Cart
5. Check-Out
6. Customer Information Entry
7. Order Confirmation

Simultaneous HTTP requests were generated to simulate 100 virtual client sessions. This simulated activity at thresholds of between approximately three to twelve million requests per day. As explained above, variations in the number of requests per day between ColdFusion Server 5, ColdFusion MX Server Enterprise Edition and ColdFusion MX for IBM WebSphere are a result of different server settings for the number of simultaneous requests allowed. Each stress test was performed multiple times to confirm accuracy. Averaged performance numbers were used for this document.

In real-world Web applications, most of the session time is spent waiting for the user to do something. A large Web site or application may have hundreds of open user sessions, while only a few dozen are actively making requests to the application server. The rest of the sessions are waiting for the user to do something.

Recreating realistic user pauses and mistakes is very difficult, so for the purposes of this test, user pauses were removed altogether. Thus, 100 simultaneous virtual user sessions does not represent 100 simultaneous users. Rather, the number of simultaneous users would be at least an order of magnitude greater. While the exact ratio of open sessions to active sessions is difficult to generalize because of the disparity in Web applications, a test using 500 virtual users would roughly translate to 10,000 open user sessions.