IBM xSeries x366 and x460 Performance Using Siebel 7.7 Enterprise Software

Mark Trbojevic
ISV Business Strategy and Enablement
February 2006

Table of Contents

Table of Figures

Abstract

This paper discusses the relative performance characteristics of the third-generation, X3 Architecture–based IBM eServer xSeries x366 and x460 systems, when Siebel 7.7 application software is used to generate enterprise quality user loads. The x366 system was tested with both single-core and dual-core CPU sets, whereas the x460 system was tested using single-core CPUs in one-chassis and two-chassis, single-partition configurations. User loads were generated using a standard benchmark kit. The Siebel 7.7 benchmark kit is developed by Siebel Engineering and supplied to all its hardware partners.

This paper will show that a 64% increase in relative CPU performance can be demonstrated by the xSeries systems when compared to an Intel 4-way, white-box server. It will also show that the xSeries systems can support greater than 200% more Siebel 7.7 users than the aforementioned Intel server, when running comparable Siebel 7.7 test workloads.

Introduction

The latest IBM® xSeries® Intel® Xeon® MP-based servers are built on the innovative new IBM X3 Architecture. IBM X3 technology offers Intel scalability that is unparalleled in the four-socket, mid-tier, application server market.

Siebel® has been selling enterprise software solutions on 4-way Intel servers for some time, but scalability has always been achieved horizontally, by adding more hardware servers to the enterprise. The third-generation x366 and x460 systems demonstrate new levels of Intel scalability that lessen the need for a customer to expand horizontally. In addition, the x460 system can, by itself, scale vertically, up to a current maximum of eight chassis.

This paper discusses the new levels of Siebel 7.7 scalability that can be achieved with an x366 or x460 system.

Software Versions

The x366 and x460 performance runs were completed using the following core software packages and versions:

- Microsoft® Windows® Server 2003 Enterprise Edition (32-bit version)
- Microsoft Internet Information Services (IIS) 6.0 (32-bit version)
- Siebel eBusiness Software 7.7.2.2 SIA [18356]
- Mercury™ LoadRunner® 7.8, Build 1406
    - Controller 7.8.0.0
    - Load Generator 7.8.0.0
    - Virtual User Generator (VuGen) 7.8.0.0
    - Analysis 7.8.1.0
- DB2 Universal Database™ (DB2® UDB) for AIX 5L™ (AIX ®) on pSeries® system Version 8.2.1
    - Code release SQL08021
    - Level identifier 03020106
    - Informational tokens DB2 v8.1.1.82, OD_14081, U803799_14081 (Fixpack 8)
- DB2 UDB for Windows Client V8.2.1
    - Code release SQL08021
    - Level identifier 03020106
    - Informational tokens DB2 v8.1.8.897, siebel_13937, WR21348_13937 (Fixpack 8)

Test Methodologies

Two distinct types of tests were run using up to four xSeries hardware configurations. Siebel 7.7 eBusiness application software and DB2 UDB V8.2.1 were used throughout all the runs.

One type of test focused on capturing sets of performance metrics from the xSeries system as increasing concurrent loads of specific Siebel users were ramped up towards the limits of the machines. These tests were collectively referred to as the **Intel Performance Study**. The xSeries hardware configurations used for these tests were as follows:

- x366 system with 4x single-core, 3.6 GHz Intel Xeon CPUs (4x 3.6 GHz CPU Cores)
- x366 system with 4x dual-core, 3.0 GHz Intel Xeon CPUs (8x 3.0 GHz CPU Cores)
- x460 system with 8x single-core, 3.3 GHz Intel Xeon CPUs (8x 3.3 GHz CPU Cores)

Another type of test focused on capturing a figure for the maximum number of concurrent, real-world workload, Siebel users that can be achieved with various xSeries hardware configuration. These tests were collectively referred to as the **Siebel Platform Sizing and Performance Program** (PSPP) **Benchmarks**. The xSeries hardware configurations used for these tests were as follows:

- x366 system with 4x single-core, 3.6 GHz Intel Xeon CPUs (4x 3.6 GHz CPU Cores)
- x366 system with 4x dual-core, 3.0 GHz Intel Xeon CPUs (8x 3.0 GHz CPU Cores)
- x460 system with 4x single-core, 3.3 GHz Intel Xeon CPUs (4x 3.3 GHz CPU Cores)
- x460 system with 8x single-core, 3.3 GHz Intel Xeon CPUs (8x 3.3 GHz CPU Cores)

Siebel 7.7 PSPP benchmark kit

All runs were completed using a Siebel 7.7 PSPP kit. The kit was developed by Siebel Engineering and then distributed in a basic form to all its hardware partners.

The benchmark kit for Siebel 7.7 was designed to run against its eBusiness Standard Industry Application (SIA) Financial Services software. It consists of the standard Siebel 7.7 SIA software, a populated database schema, two Mercury LoadRunner front-end client scripts, and an HTTP-based, back-end batch script. Siebel also included a set of guidelines and rules for running the benchmark kit, together with some Key Performance Indicators (KPIs) to which any published results must adhere.

The two front-end client scripts were designed to stress different aspects of the Siebel 7.7 Application Server:

- **Script 1: Financial Services Call Center Script**. This is the main script, which represents a typical series of transactions performed by a call center agent in response to a telephone call from a customer.

- **Script 2: eChannel Script**. This script represents a customer who is accessing a self-service Web site to gather information and request services. Although the script itself is much shorter than Script 1, it causes workflow processes to be kicked off and completed in the background. This workflow process adds significant load to the Siebel Application Server.

Each hardware partner has complete control over how it installs and configures Siebel 7.7 to support the target benchmark workloads.

Intel performance study approach

The focus of the Intel performance study was to produce a series of reproducible data points based on user count that can be compared to earlier results. In 2005, Siebel ran a performance study of Siebel 7.7 SIA application software using the best Intel 4-way, white-box server configurations that were available at the time. The focus of the Siebel study was to gain an insight into the factors that influence the performance of the Siebel Application Server running on an Intel processor-based Windows platform.

Some of the tests were run on servers installed with 4-way Intel Xeon MP processors. The core results from these tests are reproduced in **Appendix E**. These results allowed the testing team to make a direct comparison with results obtained from the new generation of x366 and x460 systems.

To be able to make any comparisons, the testing team had to capture the results using the same basic methodology that Siebel applied to its tests, as follows:

- The server must be running a Siebel-supported 32-bit Windows operating system.
- Only one instance of a Siebel 7.7 Application Server is installed on the server.
- The Siebel 7 benchmark kit must be used to drive the user load.

To comply with these conditions, Windows Server 2003 Enterprise Edition (32-bit version) was installed on an x366 system, fitted with 4x single-core, 3.6 GHz CPUs. On top of this, a Siebel 7.7.2.2 SIA Enterprise configuration was installed, consisting of a Siebel Gateway and one Siebel Application Server. The Siebel Application Server was connected to a DB2 UDB database through a supported DB2 UDB client. The database was loaded with the standard schema supplied with the Siebel 7.7 benchmark kit.

An IBM BladeCenter® HS20 blade server, loaded with Windows Server 2003 Standard Edition, was used as the Web server for the Siebel Enterprise configuration. The blade server was running Internet Information Services (IIS) 6, loaded with the standard Siebel Web Server Engine (SWSE) Plugin for Siebel 7.7.2.2.

During the Siebel study, client load was generated using a subset of the Siebel 7.7 benchmark kit. Siebel ran a Mercury LoadRunner scenario consisting of 100% **Script 1** clients. The scripts were run using PSPP default think times. The same LoadRunner scenario configuration was used for the IBM Intel Performance Study runs.

After the initial software installations, a period of time was spent performing various runs to shake out the Siebel Enterprise configuration options. Various configurations were tried and the one that produced the most stable, performance-oriented run was adopted as the base configuration for all future runs. The process of capturing an Intel Data Point run then proceeded as follows:

1. Set the LoadRunner scenario for the target number of users.
2. Configured the Siebel Application Server to support a little more than the target number of users. (As a rule of thumb, a buffer of 100 extra users was configured for every run.)
3. Restarted the Siebel Application Server and IIS Web Server, so that the testing team had a clean Siebel Enterprise configuration before each run started.
4. Started the LoadRunner scenario and allowed all users to log in and reach their steady-state[1] condition.
5. Gathered a full set of performance statistics for a continuous 15-minute period during steady state, from the Intel server running the Siebel Application Server.
6. Stopped the LoadRunner scenario and allowed all users to log off.
7. Ran the script results through Mercury LoadRunner Analysis and confirmed that the transaction times during steady state were acceptable and valid.

After a valid Intel Data Point was captured, the environment was reset for another run with an incrementally higher user count. The above process was continued until the testing team ran out of a core xSeries system resource (be that CPU capacity or available user RAM). All xSeries system performance statistics were captured using the Microsoft Performance Monitor (**Perfmon**) tool.

When the limits of the xSeries system configuration were exceeded, the configuration was changed to the next desired hardware setup and the runs started again from the beginning. A summary of the Intel Data Point study results can be found in **Appendix C**.

PSPP benchmark approach

The focus of a Siebel Platform Sizing and Performance Program (PSPP) Benchmark is very different to that of the previously discussed Intel Performance Study. A PSPP Benchmark focuses on producing the best Siebel Enterprise configuration that will yield the greatest number of real-world[2] users, on a particular hardware setup.

[1] A **steady-state** condition is when a script has run enough iterations such that all Siebel application server *transactional objects* that take part in the script, for any given Siebel user, have been initialized and called at least once. It is the point at which all necessary files are open, common data is cached, and user memory is allocated and stable.

[2] **Real-world** users are considered users that are performing typical tasks at typical transaction rates, which can be observed at any typical customer deployment of a Siebel Enterprise. Siebel surveys its customer base to determine what constitutes **real-world** workload.

A lot of time is typically spent trying to determine the limits of the hardware and software, usually through iterative tuning exercises. However, when the best configuration is found, within the constraints of the investigation time allowed, PSPP runs are captured for documentation and publication.

Usually, the primary goal of a PSPP benchmark project is to produce a figure for the total number of real-world users that comply with PSPP rules and beat all other hardware vendors. This usually involves a lot of hardware, an extensive Siebel Enterprise configuration, and many Siebel application servers.

However, the goal of these particular xSeries PSPP benchmarks was to demonstrate the extraordinary scalability of the x366 and x460 systems, as stand-alone, single-instance Siebel Application Servers. To this end, only one new-generation xSeries system took part in each benchmark run, with one instance of a front-end Siebel 7.7 application server installed.

A PSPP benchmark run uses the full Siebel 7.7 benchmark kit, requires careful planning and coordination of the tasks, and involves a lot of data capture from the multiple machines used in a run. Also, in addition to the Siebel Application Server that supports the front-end client load, an additional Siebel application server has to be added to the Enterprise configuration to support the back-end batch script load.

Three more BladeCenter HS20s, loaded with Windows Server 2003 Standard Edition, were added to the topology to support the extra software needed. One blade server was used to host the additional back-end Siebel Application Server and its associated IIS Web server with Siebel SWSE Plugin. The second blade server was used to run the Java™ HTTP back-end batch workload scripts. The third blade server was used to run IBM FastT Storage Manager software, which gathers statistics from the IBM TotalStorage® DS4300 Storage Server the testing team used to host the PSPP database containers.

After the initial software installations, a period of time was spent performing various runs to shake out the Siebel Enterprise configuration options. Various configurations were tried and the one that produced the most stable, performance-oriented run was adopted as the base configuration for all future runs. Using a mandated PSPP mix of front-end clients in an appropriate LoadRunner scenario, more runs were completed with ever-increasing user counts until the front-end application server tier reached its limit. The basic processes of capturing a PSPP benchmark run then proceeded as follows:

1. Set the LoadRunner scenario for the target number of PSPP users, using an 80:20 mix of PSPP scripts. A PSPP benchmark is mandated to include 80% Call Center users (Script 1) and 20% eChannel users (Script 2).
2. Configured the front-end Siebel Application Server to support a little more than the targeted number of users. (As a rule of thumb, a buffer of 100 extra users was configured for every run.)
3. Configured the back-end Siebel Application Server to support the required number of HTTP batch scripts.

4. Started the LoadRunner scenario and allowed all front-end users to log in and reach their steady-state condition.
5. Started capturing FastT disk statistics and DB2 UDB database server statistics.
6. Started the back-end HTTP batch scripts and allowed them to log in and reach their steady-state condition.
7. Gathered a full set of performance statistics for a continuous 30-minute period during steady state, from all the machines involved in the run.
8. Stopped the LoadRunner scenario and allowed all front-end users to log off. (The back-end HTTP batch scripts stopped themselves when a prescribed volume of transactions were completed.)
9. Ran the script results through Mercury LoadRunner Analysis and confirmed that the transaction times during steady state were acceptable and valid for a PSPP benchmark submission.

After a valid PSPP benchmark was captured with the current xSeries system configuration, the hardware was changed to the next desired configuration. The process of performing PSPP runs to find the limit of the front-end application server tier was repeated. The above process was then repeated to capture a new PSPP benchmark. A summary of the PSPP benchmark results can be found in **Appendix F**.

IBM xSeries x366 Intel System

Main Features

XA-64e 3rd-Generation Chipset
Xeon MP 32/64-bit
XceL4v Cache
PCI-X2 266 MHz
Serial Attach SCSI
Dual-Core Capable

x366: 1-way to 4-way
Rack-optimized 4-way in 3U

Comparison with xSeries Predecessor

Second Generation x365 System

- XA-32 2nd-Generation Chipset
- 1-way to 4-way, 32-bit with 400 MHz FSB
- Intel Xeon MP: 2.0/1M, 2.2/2M, 2.7/2M, 3.0/4MB
- 16 DIMMs Total: 8 Standard, 8 Optional
- 32-GB Max Memory (16 x 2-GB DIMM)
- PC2100 DDR SDRAM, 2-way Interleaving
- LSI 53C1030 Ultra320 SCSI, Integrated RAID-1
- Max Storage = 6 HDDs x 146 GB = 876 GB
- Active PCI-X: 4@133 MHz, 1@100 MHz, 1@33 MHz
- Remote I/O + RIO Sharing between x365s
- 24X CD-ROM
- Broadcom 5704 Dual Port Gigabit Ethernet
- Chipkill + Memory ProteXion + Memory Mirroring
- 2 x 950-Watt Hot-swap Power Supplies, N+N, 110V/220V
- Remote Supervisor Adapter 2 Standard
- 1-year or 3-year Next Business Day 9x5 Warranty
- 3U: 17.46”(444 mm) x 5.07”(129mm) x 28.1”(715mm)

Third Generation x366 System

- XA-64e 3rd-Generation Chipset
- 1-way to 4-way SMP, Dual-core Capable
- Dual-bus x86-64 Architecture, 667-MHz FSB
- Intel Xeon MP >3.0 GHz
- 16 DIMMs Total: 4 Standard, 12 Optional
- 64-GB Max Memory (16 x 4-GB DIMM)
- DDR2 SDRAM PC2-3200, 2-way Interleaving
- Adaptec Serial=Attached SCSI (SAS), opt. RAID5
- Max Storage = 6 2.5” HDDs x 73 GB = 438 GB
- Active PCI-X 2.0: 6 slots @ 266 MHz, No Remote I/O
- 8X DVD-ROM
- Broadcom 5704 Dual Port Gigabit Ethernet
- Chipkill + Memory ProteXion + Memory Mirroring
- 2 x 1300-Watt Hot-swap Power Supplies, N+N, 220V
- Remote Supervisor Adapter 2 Slimline opt.
- 3-year Next Business Day 9x5 Warranty
- 3U: 17.46”(444 mm) x 5.07”(129 mm) x 28.1”(715 mm)

Changes between models are highlighted in RED.

IBM xSeries x460 Intel System

Main Features

XA-64e 3rd-Generation Chipset
Xeon MP 32/64-bit
XceL4v Cache
PCI-X2 266 MHz
Serial Attach SCSI
MXE-460 (Scalability)
Dual-Core Capable

x460: 2-way to 32-way
Integrated 4-way rack in 3U
Scalable to 8-chassis, 32-way

Comparison with xSeries Predecessor

Second Generation x445 System

- XA-32 2nd Generation Chipset
- 2-way to 32-way SMP, 32-bit with 400-MHz FSB
- Intel Xeon MP: 2.0/1M, 2.2/2M, 2.7/2M, 3.0/4MB
- Intel Xeon DP 3.0GHz up to 4-way
- 64-MB XceL4 per CEC, 512-MB Max
- 64-GB max-addressable memory supported
- DDR SDRAM PC2100, 2-way Interleaving
- LSI Ultra320 SCSI, Integrated RAID-1
- Max Storage = 2 HDDs x 146 GB = 292 GB
- Active PCI-X: 2@133 MHz, 2@100 MHz, 2@66 MHz
- Remote I/O + RIO Sharing
- 2 x 1200-Watt Power Supplies, Hot-swappable
- Broadcom 5704 dual-port GbE
- Remote Supervisor Adapter II for EXA (std)
- Active Memory + Hot-swap and Hot-add Memory
- 3-year Next Business Day 9x5 Warranty
- 4U: 19”(483 mm) x 7”(178 mm) x 28.1”(714 mm)
- 20 #1 Benchmarks…and counting!!!

Third Generation x460 System

- XA-64e 3rd-Generation chipset
- 2-way to 32-way SMP, Dual-core Capable
- Dual-bus x86-64 Architecture, 667-MHz FSB
- Intel Xeon MP: 2.83/4M, 3.16/8M, 3.33/8MB
- 256-MB XceL4v per chassis, 2 GB Max
- 64-GB Max-Memory per chassis, 512 GB Max Total
- DDR2 SDRAM PC2-3200, 2-way Interleaving
- Adaptec Serial Attached SCSI (SAS), Opt. RAID5i
- Max Storage = 6 2.5” HDDs x 73 GB = 438 GB
- Active PCI-X 2.0: 6 available slots, all 266 MHz
- MXE-460 Modular Xpansion Enclosure (>4-way)
- 2 x 1300-Watt Power Supplies, Hot-swappable
- Broadcom 5704 dual port GbE
- Remote Supervisor Adapter II Slimline (std)
- Active Memory + Hot-swap (All DIMMs accessible)
- 3-year Next Business Day 9x5 Warranty
- 3U: 19”(483 mm) x 5.25”(133 mm) x 27.5”(698mm)

- Up to 125% performance improvement over x445 system

Changes between models are highlighted in RED.

IBM xSeries X3 Architecture

The IBM xSeries X3 Architecture represents the latest delivery of the IBM advanced Enterprise X-Architecture™ technology (EXA2).

Background

In 2003, IBM was the first in the industry to release the second generation of innovative Intel processor server platforms; the advanced Enterprise X-Architecture technology (EXA2).

IBM X-Architecture technologies give xSeries systems their name and mainframe-like stability. In combination with the line's extraordinary management tools, these techniques set xSeries systems apart from other Intel processor-based systems.

In 2005, IBM again extended its performance leadership in the 4-socket, Intel processor server industry with the release of the X3 Architecture in a new line of xSeries systems. The x366 and x460 systems are the flagships of the third-generation machines and deliver breakthrough 32-bit and 64-bit performance:

- The x366 system secured a new number-1, single-core, TPC-C database benchmark with a result that showed 38% greater performance than a previous generation x365 4-way system.

- The x460 system produced a new number-1, single-core, TPC-C database benchmark with a result that showed 61% greater performance than the previous generation x445 8-way system. In fact, the x460 8-way system had 16% greater performance than a previous generation x445 16-way system.

Processor performance issues

Clocks per Instruction (CPI) is a key metric used by processor and system designers to identify the performance efficiency of a processor. The metric is somewhat analogous to the miles-per-hour and miles-per-gallon metrics used to judge an automobile’s performance.

In the computing world, a lower figure for CPI is desirable and a processor is at its most efficient when all instructions and data are resident in the processor’s fastest cache memory. This idealized measure is known as Infinite Cache CPI or Core CPI.

However, in a real-world system, processor fast cache is finite and all instructions and data cannot fit in there. A processor will have to go to off-chip memory for data. Frequent cache misses take much longer to service than a cache hit. A processor must wait a longer period of time before obtaining data or instructions from real memory. This increases the average CPI of a processor that is running real-world applications.

Processor performance can be improved dramatically by reducing the average number of processor clocks that are needed to process instructions (that is, Core CPI). However, you can only go so far with Core CPI improvements. Improving Core CPI only affects the time spent inside a processor. With real-world systems, this time is usually much smaller than the total time that is spent waiting on the system outside the processor.

The external component of CPI above and beyond the Core CPI is where processor performance is lost. This external component is made up of chipset and bus latency. IBM X3 technology greatly improves performance and scaling by focusing on reducing external CPI (that is, latency).

Introduction to X3 technology

The X3 Architecture has been designed from the ground up with ultra-low memory latency to provide optimal performance for multiuser, multithreaded commercial application workloads. It represents innovation that is focused on the following performance and features:

- Imbedded eDRAM snoop filter that improves the performance of:
  - Front-Side Bus (FSB) operations
  - PCI throughput
  - Multinode scalability
- Dramatic processor-to-memory latency reductions
- Vastly improved I/O performance
- Twin 667 MHz Front-Side-Buses, isolated to reduce bus contention:
  - Without any non-uniform memory access (NUMA) latency in 4-way and smaller systems
  - Optimal performance can be obtained without NUMA-aware software

Advanced Enterprise X-Architecture (EXA2) versus X3 Architecture

The following series of diagrams (Figure 1 through Figure 4) illustrate the advancements made between the IBM second-generation EXA2 technology and the new third-generation X3 Architecture.

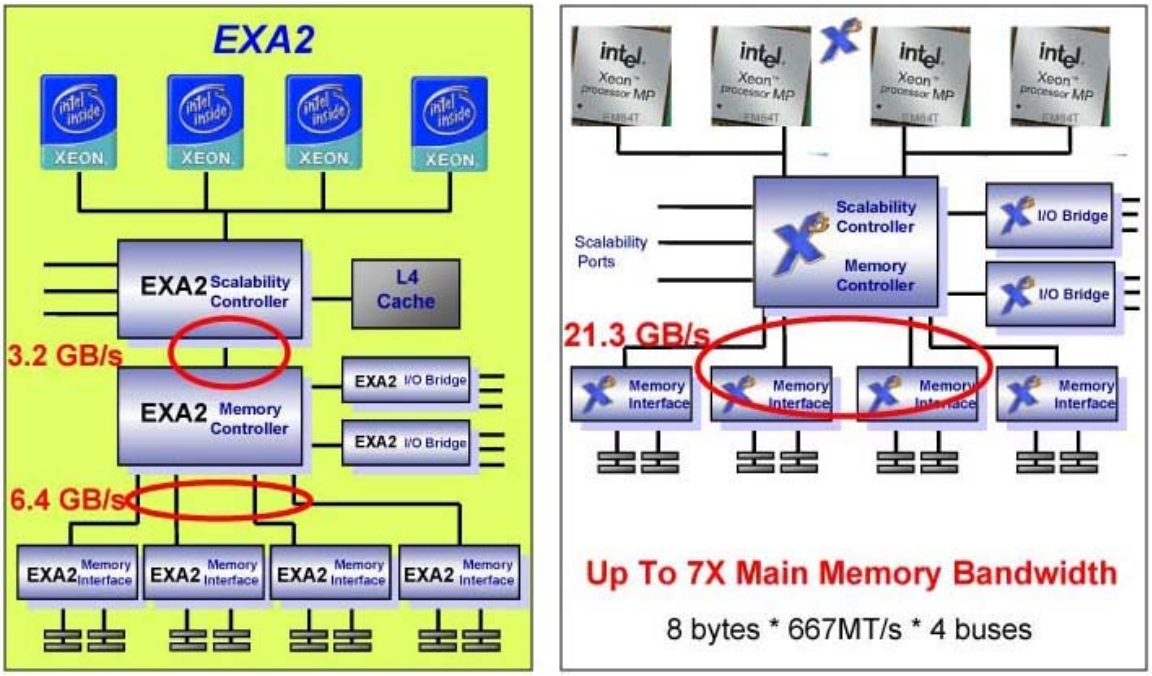


Figure 1: Greater Main Memory Bandwidth

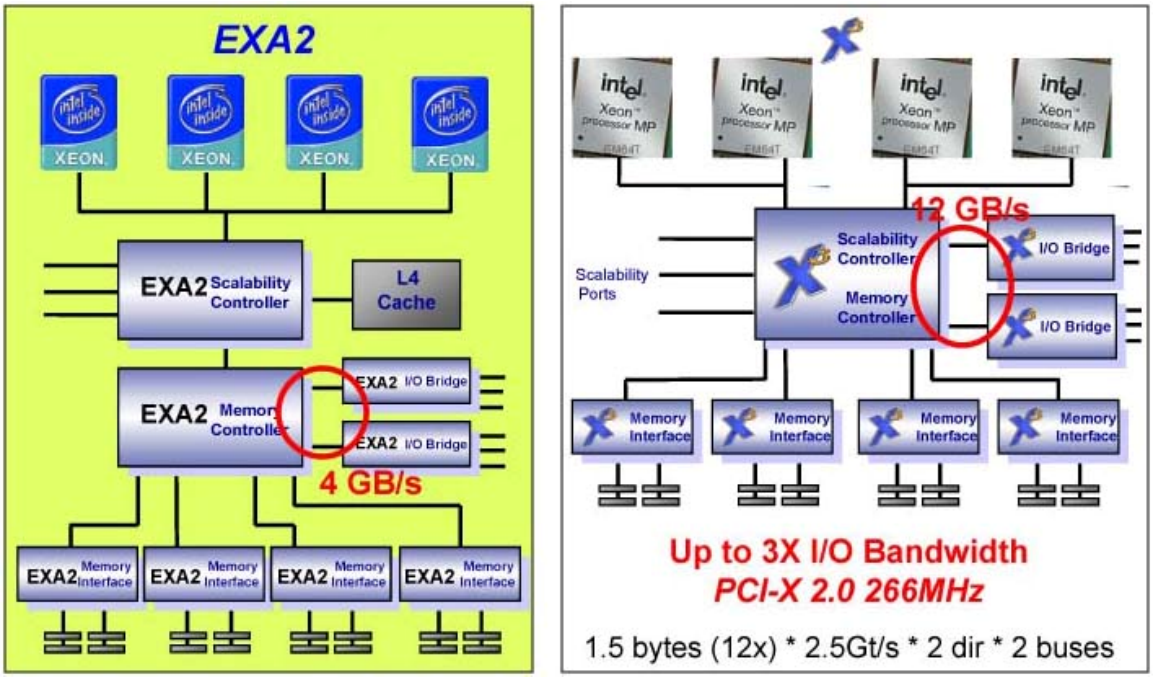


Figure 2: Improved I/O Bandwidth

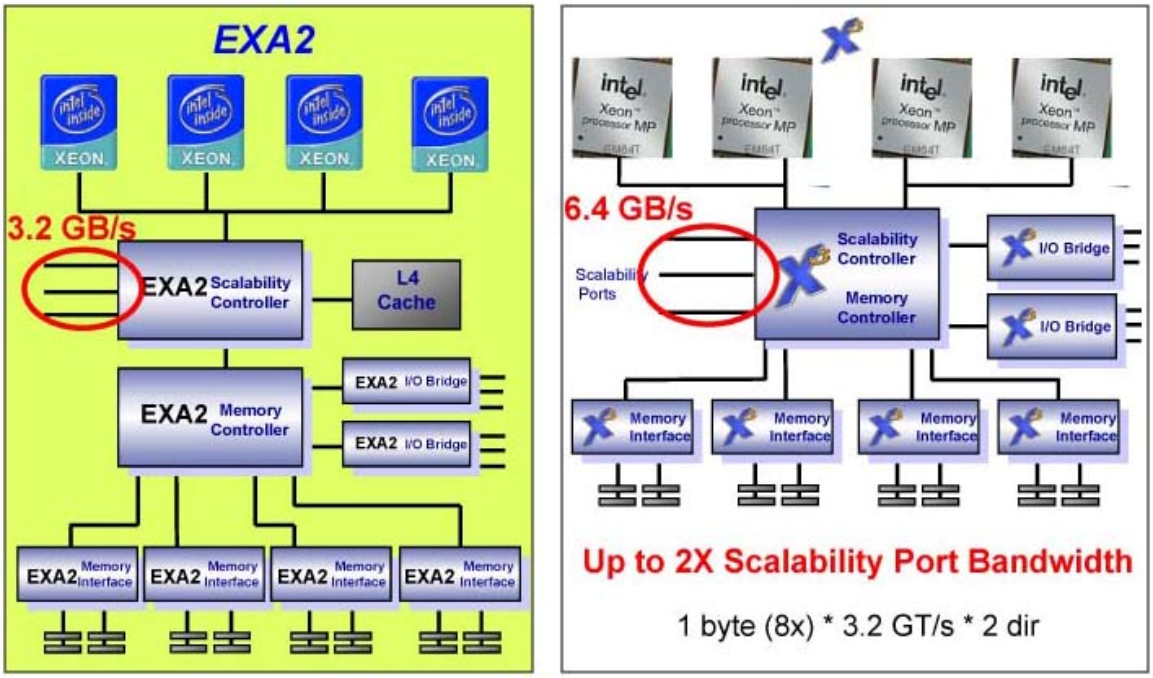


Figure 3: Improved Scalability Port Bandwidth

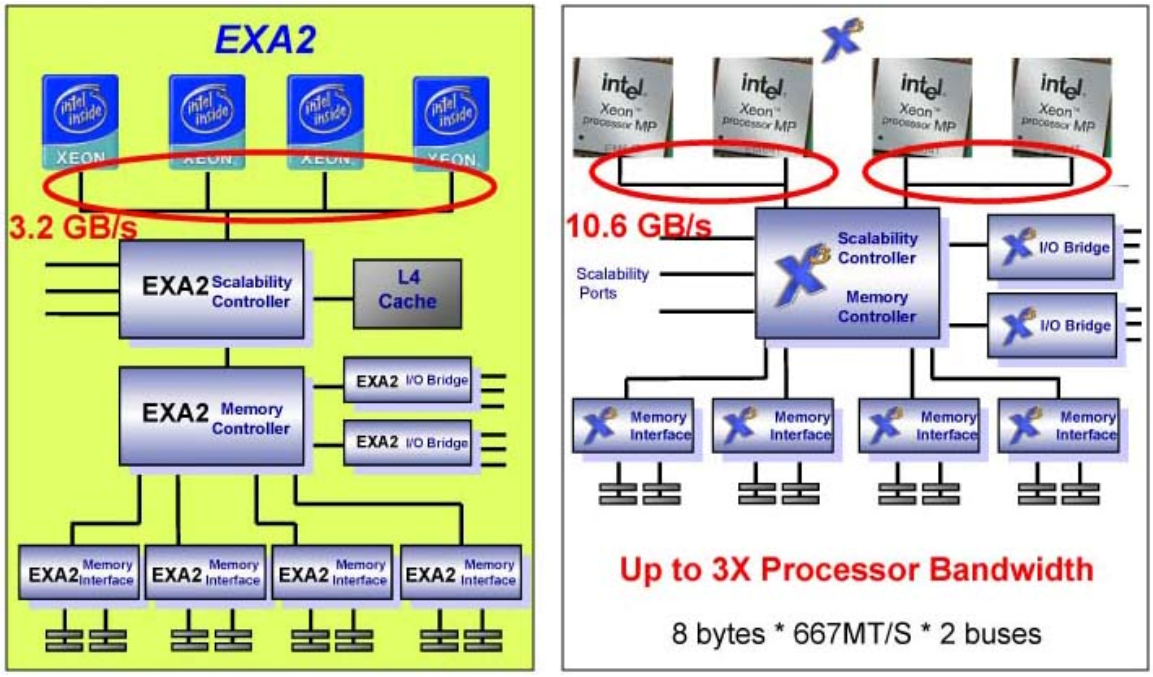


Figure 4: Improved Processor Bandwidth

Software installation and configuration

The diagrams in Figure 5 and Figure 6 illustrate the software installation and configuration details for the benchmarks.

Hardware topology for Intel Performance Study runs

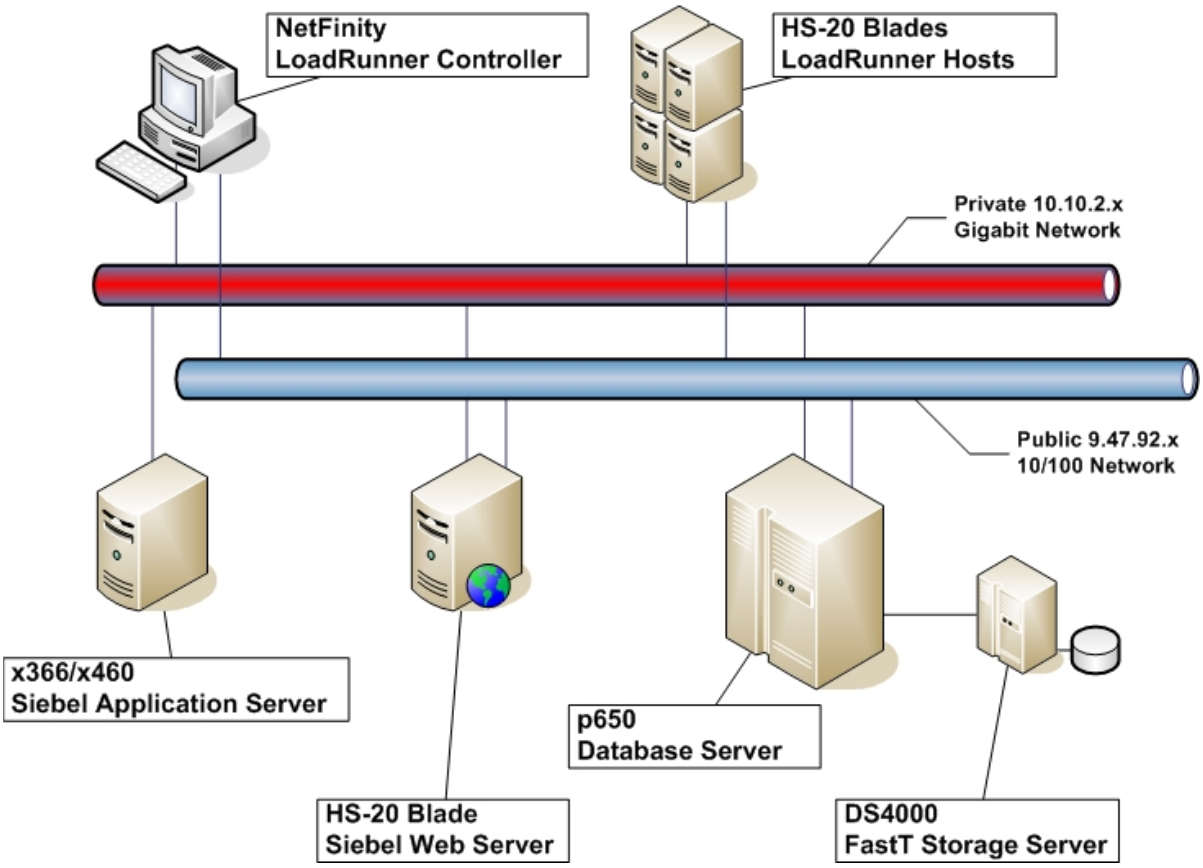


Figure 5: Network infrastructure for Intel Performance Study

Hardware topology for PSPP Benchmark runs

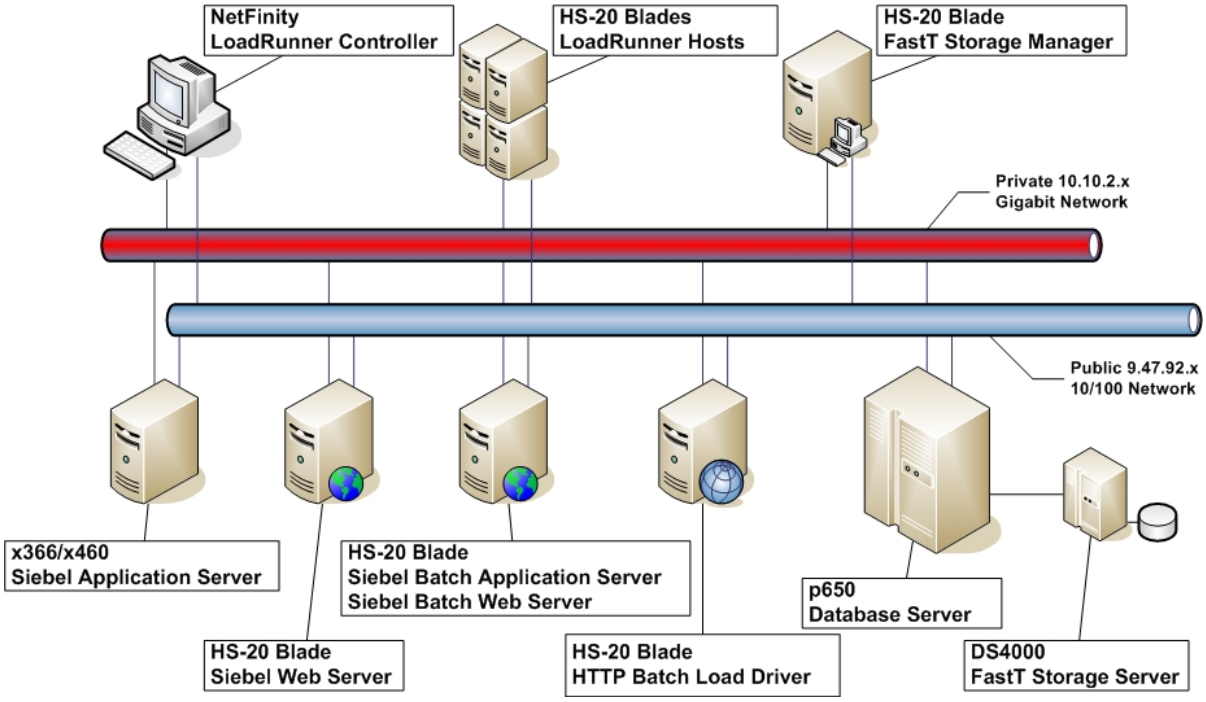


Figure 6: Network infrastructure for PSPP Benchmark

IBM x366 and x460 system application tier

The application tier for the x366 and x460 systems is explained below.

Initial Setup

The first sets of runs were completed with an x366 system fitted with 4x single-core, 3.6 GHz Intel Xeon MP CPUs. The x366 system was fitted with four memory cards over which 8x 2 gigabyte (GB) DIMMs and 8x 1GB DIMMs were spread. A performance-sensitive memory configuration was adopted, with the 2GB DIMMs installed in slots one and three on each card and the 1GB DIMMs in slots two and four. This gave the x366 system a 24GB RAM capacity.

A single 36GB hard disk was fitted in the x366 system. All microcode and BIOS software was brought up to the latest available versions. All BIOS options were left at their default settings, with the following two exceptions:

- Under **Advanced Setup - CPU Options**, the **Hardware Prefetch** option was set to **disabled.** (This was determined, through empirical testing, to be the best setting for a Siebel Application Server. However, the actual performance gained by disabling this setting was barely noticeable and cannot be regarded as a significant tuning option for the Siebel Application Server.)

- Under **Advanced Setup – Memory Options**, the x366 system was configured for **High Performance Memory Array** use. (This was deemed the optimal setting to use with the DIMM configuration that was installed. No tests were completed with other BIOS memory settings because all the other options sacrificed total RAM availability for greater stability and reliability. Siebel is a very memory-intensive application and the setting chosen here guarantees the fastest possible access to the installed RAM.)

Windows Server 2003 Enterprise Edition (32-bit version) was installed as the operating system on the single hard drive. Siebel is not compiled for, nor certified on any Windows 64-bit operating system. The x366 system was connected to two networks in the lab; a private gigabit network and a public 10/100 network (see Figure 5 and Figure 6). The private network was used for all application related traffic, whereas the public network was used for control and maintenance functions.

After network connectivity to the configured database server was confirmed, a DB2 UDB V8.2.1 database client was installed and configured. This was followed by the Siebel 7.7 software. The Siebel Enterprise Gateway was installed together with the Siebel Application Server on the x366 boot disk. (Every Siebel Enterprise configuration needs a Gateway, but the Gateway in Siebel 7.7 is merely a name server that stores configuration information. It is accessed when components of the Siebel Enterprise configuration start up, but takes no active role during a run.)

Software Optimization

With all of the hardware and software in place, many baseline runs were completed to determine the best configuration options. The initial runs were

focused on Windows configuration options and after the best Windows options were determined, the remaining runs focused on the best Siebel configuration.

Previous benchmarks and Windows configuration best practice guides were consulted to determine the most appropriate Windows options to modify. In the end, the testing team determined that the most efficient Windows configuration for Siebel 7.7 software was obtained when Windows was left to configure itself. Most manual changes to the Windows operating system had either a detrimental effect or no significant effect on Siebel Enterprise performance.

Only one manual configuration change was left in place after testing and that was to run with no system page file. This was done partly because there was a significant amount of physical RAM available and partly to save disk space on the single available hard drive. Even so, running with or without a system page file had no significant effect on Siebel Enterprise performance.

Hardware compatibility

A convenient feature of the x366 and x460 system is that internally, they are built from the same parts catalog. This means that core components can be changed and interchanged between the systems without having to completely reinstall or reconfigure all software. Windows Server 2003 is a plug-and-play compliant operating system. It is capable of recognizing significant hardware changes and reconfiguring itself accordingly. The testing team was able to take advantage of this during the performance tests.

Four distinct xSeries hardware configurations were used for the Siebel 7.7 Application Tier during the performance tests, as follows:

1. x366 system:
    - 4x single-core 3.6 GHz Intel Xeon MP CPUs (4x 3.6 GHz CPU Cores)
    - 4x memory cards (4x DIMM slots per card)
    - 24GB of RAM
        - 8x 2GB DIMMs installed in slots one and three
        - 8x 1GB DIMMs installed in slots two and four
    - 1x 36GB hard disk (boot disk)

2. x366 system:
    - 4x dual-core, 3.0 GHz Intel Xeon MP CPUs (8x 3.0 GHz CPU Cores)
    - 4x memory cards (4x DIMM slots per card)
    - 24GB of RAM
        - 8x 2GB DIMMs installed in slots one and three
        - 8x 1GB DIMMs installed in slots two and four
    - 1x 36GB hard disk (boot disk)

3. x460 system (single-chassis setup):
    - 4x single-core 3.3 GHz Intel Xeon MP CPUs (4x 3.3 GHz CPU Cores)
    - 4x memory cards (4x DIMM slots per card)
    - 16GB of RAM
        - 8x 2GB DIMMs installed in slots one and three
    - 1x 36GB hard disk (boot disk)

4. x460 system (two-chassis, single-partition setup):
    - 2x 2.3m length scalability cables
    - 8x single-core, 3.3 GHz Intel Xeon MP CPUs (8x 3.3 GHz CPU Cores)
        - 4x CPUs in x460 chassis
        - 4x CPUs in MXE 460 chassis
    - 8x memory cards (4x DIMM slots per card)
        - 4x cards in x460 chassis
        - 4x cards in MXE 460 chassis
    - 32GB RAM
        - 8x 2GB DIMMs installed in slots one and three in x460 chassis
        - 16x 1GB DIMMs installed in slots one and three, as well as in slots two and four in MXE 460 chassis
    - 1x 36GB hard disk (boot disk)
        - installed in x460 chassis

To change from hardware configuration (1) to (2), the x366 system was powered down and the single-core CPU board was replaced with a new dual-core CPU board. When the new configuration was powered up, Windows Server 2003 recognized the different CPUs and reconfigured the operating system accordingly. All the installed software on the boot disk was unaffected by the hardware changes.

To change from hardware configuration (2) to (3), the x366 system was again powered down. The boot disk from the x366 system was fitted into the first drive bay on the x460 chassis. The memory cards were removed from the x366 system, the 1GB DIMMs were removed from the cards, and the cards were fitted into the x460 chassis. All the cables from the x366 system were reconnected to the same positions on the x460 chassis. When the x460 system was powered up, Windows Server 2003 recognized all the changed hardware and reconfigured itself accordingly. All the installed software on the boot disk was unaffected by the hardware changes. All the basic Windows operating system configuration settings, such as network settings, were retained.

To change from hardware configuration (3) to (4), the MXE 460 chassis was added to the x460 system. The x460 system was powered down. The 1GB DIMMs removed from the x460 chassis were added to the existing 8x 1GB DIMMs in the MXE 460 system. The MXE 460 system was cabled by two scalability cables to the x460 chassis. Using the Remote Supervisor Adapter (RSA) II Web interface on each machine, the x460 chassis and the MXE 460 chassis were configured into a single-partition x460 system. When the x460 system was powered up, the two chassis merged into one partition. Windows Server 2003 recognized the additional resources and reconfigured itself

accordingly. All the installed software on the boot disk was unaffected by the hardware changes.

In each hardware configuration, all microcode and BIOS software versions were maintained at the latest available levels. All BIOS settings were kept the same when the switch was made from the x366 system to the x460 system. The BIOS settings of the MXE 460 system were set to match the x460 chassis when it was added to the x460 partition. CPU hyperthreading was enabled during every run. See **Appendix A** and the **Appendix B** for a high level summary of the Windows operating system configurations that were active during each hardware incarnation.

Software settings

During both the Intel performance study and the PSPP benchmark runs, the focus was on x366 and x460 system performance, and how well the Siebel 7.7 Application Server scales on these platforms. Time was taken to find a Siebel 7.7 Application Server configuration that yielded the best results in terms of CPU and memory utilization. After being discovered, this base configuration was not altered as the application server was reconfigured to support more hardware.

Examples of the Siebel Application Server configurations that yielded the best runs in each hardware category can be found in **Appendix D** and **Appendix G**.

Web server tier

The setup for the Web server tier is explained below.

Initial setup

It is common practice in Siebel projects such as this to co-locate the Siebel application server with its corresponding Web server. This type of configuration optimizes the TCP/IP communications between the Web server and the application server and reduces network traffic. However, the Web server, with its loaded Siebel Web Server Engine (SWSE) Plugin, will consume valuable system resources on the platform.

With the focus of the Intel performance study and PSPP benchmark runs being on x366 and x460 system performance, and how well the Siebel 7.7 Application Server scales on these platforms, it was important to make sure that the Siebel 7.7 Application Server had access to the maximum amount of system resources available.

To this end, a BladeCenter HS20 was used as the host for the Web server that would drive load against the x366 and x460 systems hosting the Siebel 7.7 Application Server. The BladeCenter HS20 was fitted with 2x 3.0 GHz CPUs and 4GB of RAM. Windows Server 2003 Standard Edition (32-bit version) was installed as the operating system on the HS20 blade.

Internet Information Services (IIS) 6.0 was installed and configured as the Web server. The standard Siebel 7.7 SWSE Plugin was installed on top of IIS and configured to use the private network for both client and server connections.

Software optimization

Standard Siebel 7.7 best practices were followed to configure IIS and the Siebel SWSE Plugin. Previous benchmarks and Windows configuration best-practice guides were consulted to determine the most appropriate Windows options to modify. In the end, it was again determined that the most efficient Windows configuration for IIS and the Siebel SWSE was obtained when Windows was left to configure itself. None of the Windows optimization changes had any significant effect on Siebel Enterprise performance.

Software settings

After the initial setup of the BladeCenter HS20 system with IIS and the Siebel 7.7 SWSE Plugin, no further software changes needed to be made. The Siebel SWSE Plugin for IIS is able to scale with IIS as far as the system resources will allow. The 2x 3.0 GHz CPUs and 4GB of RAM on the BladeCenter HS20 system were more than enough to support the largest runs that were attempted on the x366 and x460 systems.

Batch application tier

The batch application tier is explained below.

Initial setup

A PSPP benchmark run involves more software than is needed for the Intel performance study runs. An additional Siebel 7.7 Application Server and associated Web server are required to run batch workloads. This batch workload does not directly affect the Siebel client application server running on the x366 and x460 systems. Instead, it generates additional database load, which in turn, can affect the database query times observed by the client application server.

To support the PSPP batch workload, an additional BladeCenter HS20 system was added to the topology (see Figure 6). The HS20 blade server had the same hardware configuration as that used for the client Web server tier (that is, 2x 3.0 GHz CPUs and 4GB of RAM). Windows Server 2003 Standard Edition (32-bit version) was installed as the operating system on the HS20 blade server.

Unlike the Siebel 7.7 client application server on the x366 and x460 systems, the batch application server does not need to be scaled to its limits. It merely has to provide enough batch throughput to satisfy PSPP benchmark requirements, relative to the client workload. For these PSPP benchmark runs, this is well within the limits of the BladeCenter HS20 system. As a result, the testing team was able to co-locate the Siebel Application Server and its corresponding Web server on the same BladeCenter HS20 system.

Internet Information Services (IIS) 6.0 was installed and configured as the batch Web server. The standard Siebel 7.7 SWSE Plugin was installed on top of IIS and configured to use the private network for all connections. A Siebel 7.7 Application Server was installed on the same BladeCenter HS20 system and added to the existing Siebel Enterprise configuration. The SWSE Plugin was configured to communicate with the local batch application server.

Software optimization

Standard Siebel 7.7 best practices were followed to configure IIS, the Siebel SWSE Plugin, and the Siebel batch application server. Because the batch workload required for a valid PSPP benchmark submission of this scale is not very demanding, the Windows operating system was left to configure its own optimal settings.

Software settings

After the initial setup of the BladeCenter HS20 system for the batch workload, no further software changes needed to be made. PSPP batch workload is measured in terms of throughput as a function of the number of running client sessions. The client workload generated during these PSPP runs was relatively small compared to a more traditional high-water PSPP benchmark. As a result, the testing team was able to configure a fixed batch workload throughput, which was high enough for all the PSPP client runs that followed.

Database tier

The database tier is explained below.

Database size

Both the Intel performance study and the PSPP benchmark relied on a database tier that housed the repository data for the Siebel eFinance application. The PSPP benchmark DB2 UDB V8.2.1 database requires approximately 350 gigabytes of disk storage for run-time operation. The size of the database used was approximately 230 gigabytes. It was built to simulate customers with large transaction volumes and data distributions representing the most common customer data shapes. See Figure 7 for a sampling of record volumes for key business entities of the Siebel Industry Application (SIA) volume database.

Business Entity	Number of Records
Accounts	2,233,637
Activities	6,685,419
Addresses	3,475,662
Contacts	68,062
Employees	30,000
Opportunities	3,429,952
Orders	496,909
Products	230,102
Quote Items	1,984,252
Quotes	253,693
Service Requests	5,651,814

Figure 7: Base Record Counts from Core PSPP Database Tables

Database configuration

The database resided on an IBM eServer POWER4+™ processor-based eServer pSeries Model 650 with 32GB of memory. The operating system was AIX 5L V5.2 ML06 in 64-bit kernel mode. An IBM TotalStorage DS4300 Model 1722-60U was attached to the p650. The database was spread across 48x 10,000 RPM disk drives of the DS4300. DB2 UDB V8.2.1 fixpack 8 was installed in 64-bit mode.

The database was configured with additional buffer pool storage. The storage provided a database cache-hit ratio of 99%, leaving the database with acceptable I/O performance (regardless of the workload generated by the clients). See Figure 8 for buffer pool usage statistics.

Buffer Pool Name	Number of Pages	Page Size in Bytes	Memory Usage in Megabytes
IBMDEFAULTBP	70,000	4,096	273
BUF_16K	45,000	16,384	703
BUF_IDX16K	80,000	16,384	1,250
BUF_32K	5,000	32,768	156
		Total:	2,382

Figure 8: PSPP Database buffer pool statistics

The database machine ran without congestion or any significant load-bearing periods. It was set up to provide ample growth and never to be a bottleneck within the system. To give an idea of approximately how utilized the database machine was, the salient statistics for the largest PSPP benchmark run are reported in Figure 9.

CPU Usage	70%
Memory	14.7 GB
Threads	3700

Figure 9: Database usage statistics for the 3,600 users PSPP run

Other salient configuration parameters required to reproduce the results in this paper can be found in **Appendix H**.

Storage tier

The IBM TotalStorage DS4300 is a midlevel disk system that can scale to more than 16 terabytes (TB) of fibre-channel disk physical capacity using three DS4300 EXP710 Express switched expansion units. It can also scale to more than 33 TB of fibre-channel disk physical capacity with the **Turbo** feature, using seven DS4300 EXP710 Express units. It uses the latest in storage networking technology to provide an end-to-end, 2 gigabytes-per-second fibre-channel solution. As part of the DS4000 midrange disk systems family, the DS4300, and DS4300 with the **Turbo** feature use the same common storage management software and high performance hardware design, providing clients with enterprise-like capabilities found in midrange to high-end models, but at a much lower cost.

The DS4300 attached to the model p650 system had a smaller configuration. Only 48 disk drives were configured; the remaining disk drives were kept as spare drives. The DS4300 never showed any I/O stress during the benchmarks. Figure 10 shows the configuration summary of the DS4300 used.

SAN	IBM TotalStorage DS4300
Model	1722-60U
Quantity	1
Cache	896 MB
Quantity Host Adapters	2 Fibre Channel Adapters
Disk Size	56x 136GB – 10K RPM (4 drawers)
Disk Configuration	4x (12-disk) 128KB, striped logical drives

Figure 10: TotalStorage DS4300 System Summary

Load generation tier

The load generation tier is explained below.

Initial setup

The Siebel 7.7 PSPP benchmark kit comes with two Mercury LoadRunner scripts that generate client workload. The scripts rely on a Windows DLL, developed by Siebel Engineering, to parse Siebel data out of the HTTP data streams. This means that the scripts can only be executed by LoadRunner Hosts that are running on Windows platforms.

An IBM xSeries Netfinity® system was used as the LoadRunner Controller. Windows Server 2003 Standard Edition (32-bit version) was installed as the operating system on the Netfinity system. Mercury LoadRunner 7.8 Controller, Virtual User Generator (VuGen) and Analysis were installed next.

The BladeCenter HS20 systems were used as LoadRunner Hosts. Ten HS20 blades were prepared with the Windows Server 2003 Standard Edition (32-bit version) operating system. Mercury LoadRunner 7.8 Host software was installed on each HS20 blade server. Each HS20 blade server was connected to both the private and public networks. The public network was used to register each LoadRunner Host with the LoadRunner Controller. The private network was used by each script to connect to the Web server supporting the x366 and x460 system Siebel 7.7 Application Servers.

In addition to the LoadRunner client workload scripts, the PSPP benchmark kit includes some Java scripts that generate batch workload. These scripts do not require LoadRunner to operate. They just need a platform with Java installed and a network over which they can reach the Web server that is driving the Siebel 7.7 batch application server.

An additional BladeCenter HS20 system was used to drive the Java batch workload scripts. Windows Server 2003 Standard Edition (32-bit version) was installed as the operating system and JavaSoft™ 2 Standard Edition Runtime Environment (JRE) was installed to support the batch scripts.

Software optimization
No specific optimization changes were made to the LoadRunner infrastructure. None were necessary. During the Intel performance study and PSPP benchmark runs, the infrastructure was monitored for potential bottlenecks, but the client workload was well within the capacity of the LoadRunner servers and available network bandwidth.

No optimization changes were necessary on the BladeCenter HS20 system that was driving the Java batch workload either.

Software settings
As the Intel performance study and PSPP benchmark runs progressed, the only changes that needed to be made to the LoadRunner infrastructure were to the LoadRunner scenario that was driving the client workload.

The Java batch workload scripts were only used during the PSPP benchmark runs. After the configuration that generated the required batch workload throughput was determined, it did not need to be changed. It simply had to be re-run, when necessary.

Intel performance study

The results of this performance study are shown below.

Siebel results

In 2005, Siebel ran a performance study of Siebel 7.7 application software using the best Intel 4-way white-box server configurations that were available at the time. Three of the runs were completed with servers installed with Intel Xeon MP CPUs.

The Siebel study was concerned with finding the factors that effect Siebel scalability on Intel servers. However, the core data captured for the three Intel P-IV Xeon server configurations allowed the testing team to make a direct performance comparison with the x366 and x460 system results.

With permission from Siebel, the core Intel 4-way white-box data has been reproduced in Figures 11 through 13. The results are color-coded to allow easier identification of each result set in the tables and graphs in Figure 17 through Figure 21 that follow later in this section.

Intel P-IV 4-Way (4x 2.8 GHz Cores), 2MB L3 (HT OFF)

Users	400	500	600	700	800	900	1000	1100
No. of CPUs	4	4	4	4	4	4	4	4
No. of CPU Cores	4	4	4	4	4	4	4	4
Total CPU%	25.83	32.68	39.64	46.70	54.30	62.16	69.50	77.85
User CPU%	23.67	30.19	36.69	43.30	50.36	57.60	64.50	72.12
System CPU%	2.17	2.49	2.95	3.42	3.94	4.56	5.02	5.73
Hits/Sec	50.00	63.00	75.00	87.00	100.00	112.00	125.00	138.00
Avg Response	0.09	0.09	0.09	0.09	0.10	0.10	0.11	0.12
Users/CPU Core (@100%)	387	382	378	375	368	362	360	353
Users/CPU Core (actual)	100	125	150	175	200	225	250	275

Users	1200	1300
No. of CPUs	4	4
No. of CPU Cores	4	4
Total CPU%	86.20	93.03
User CPU%	80.10	86.60
System CPU%	6.10	6.46
Hits/Sec	150.00	161.00
Avg Response	0.15	0.21
Users/CPU Core (@100%)	348	349
Users/CPU Core (actual)	300	325

Figure 11: Intel P-IV 4-Way (4x 2.8 GHz Cores), 2MB L3 (HT OFF)

Intel P-IV 4-Way (4x 2.8 GHz Cores), 2MB L3

Users	500	600	700	800	900	1000	1100	1200
No. of CPUs	4	4	4	4	4	4	4	4
No. of CPU Cores	4	4	4	4	4	4	4	4
Total CPU%	18.42	22.77	27.77	33.00	38.60	45.93	53.55	62.17
User CPU%	16.93	21.19	25.70	30.70	35.89	42.78	49.85	57.94
System CPU%	1.49	1.58	2.00	2.31	2.81	3.15	3.70	4.23
Hits/Sec	63.00	75.00	87.00	100.00	112.00	125.00	137.00	150.00
Avg Response	0.10	0.10	0.10	0.10	0.10	0.11	0.12	0.13
Users/CPU Core (@100%)	679	659	630	606	583	544	514	483
Users/CPU Core (actual)	125	150	175	200	225	250	275	300

Users	1300	1400	1500
No. of CPUs	4	4	4
No. of CPU Cores	4	4	4
Total CPU%	70.47	81.22	93.23
User CPU%	65.90	75.93	87.06
System CPU%	4.57	5.38	6.17
Hits/Sec	162.00	174.00	184.00
Avg Response	0.14	0.17	0.33
Users/CPU Core (@100%)	461	431	402
Users/CPU Core (actual)	325	350	375

Figure 12: Intel P-IV 4-Way (4x 2.8 GHz Cores), 2MB L3

Intel P-IV 4-Way (4x 2.8 GHz Cores), 4MB L3

Users	500	600	700	800	900	1000	1100	1200
No. of CPUs	4	4	4	4	4	4	4	4
No. of CPU Cores	4	4	4	4	4	4	4	4
Total CPU%	16.70	20.41	24.53	29.27	34.38	40.03	45.99	52.54
User CPU%	15.48	19.00	22.89	27.35	32.05	37.33	42.89	49.09
System CPU%	1.21	1.42	1.66	1.92	2.33	2.69	3.10	3.45
Hits/Sec	63.00	75.00	88.00	100.00	113.00	125.00	138.00	150.00
Avg Response	0.10	0.10	0.10	0.10	0.10	0.11	0.12	0.12
Users/CPU Core (@100%)	749	735	713	683	654	625	598	571
Users/CPU Core (actual)	125	150	175	200	225	250	275	300

Users	1300	1400	1500
No. of CPUs	4	4	4
No. of CPU Cores	4	4	4
Total CPU%	58.96	66.67	74.61
User CPU%	55.22	62.47	70.00
System CPU%	3.73	4.20	4.61
Hits/Sec	162.00	175.00	187.00
Avg Response	0.12	0.14	0.14
Users/CPU Core (@100%)	551	525	503
Users/CPU Core (actual)	325	350	375

Figure 13: Intel P-IV 4-Way (4x 2.8 GHz Cores), 4MB L3

IBM results

Following the same testing methodology applied by Siebel to capture the Intel processor 4-way results, Figures 14 through 16 reproduce the results obtained with the x366 and x460 systems. The results are color-coded to allow easier identification of each result set in the tables and graphs in Figures 17 through 21 that follow later in this section.

x366 System 4-Way (4x 3.6 GHz Cores), 1MB L2

Users	500	700	900	1100	1300	1500	1700	1900
No. of CPUs	4	4	4	4	4	4	4	4
No. of CPU Cores	4	4	4	4	4	4	4	4
Total CPU%	9.70	13.76	17.99	25.03	29.72	37.21	42.23	51.91
User CPU%	9.01	12.72	16.72	22.98	27.43	34.38	38.98	47.86
System CPU%	0.68	1.04	1.27	2.05	2.28	2.82	3.24	4.04
Hits/Sec	58.80	82.90	106.76	129.55	153.33	176.96	201.09	224.41
Avg Response	0.04	0.05	0.05	0.05	0.05	0.05	0.06	0.06
Users/CPU Core (@100%)	1289	1272	1251	1099	1094	1008	1006	915
Users/CPU Core (actual)	125	175	225	275	325	375	425	475
RAM/User (MB)	7.07	6.44	6.12	5.93	5.80	5.68	5.59	5.55

Users	2100	2300	2500	2700	2800
No. of CPUs	4	4	4	4	4
No. of CPU Cores	4	4	4	4	4
Total CPU%	58.11	70.71	78.83	89.74	95.10
User CPU%	53.67	65.48	72.90	83.08	87.89
System CPU%	4.44	5.22	5.92	6.66	7.21
Hits/Sec	249.15	272.38	295.48	318.21	328.60
Avg Response	0.06	0.07	0.07	0.11	0.18
Users/CPU Core (@100%)	904	813	793	752	736
Users/CPU Core (actual)	525	575	625	675	700
RAM/User (MB)	5.41	5.41	5.40	5.39	5.45

Figure 14: x366 system 4-Way (4x 3.6 GHz Cores), 1MB L2

x366 System 4-Way (8x 3.0 GHz Cores), 2MB L2

Users	500	700	900	1100	1300	1500	1700	1900
No. of CPUs	4	4	4	4	4	4	4	4
No. of CPU Cores	8	8	8	8	8	8	8	8
Total CPU%	5.56	7.80	10.64	12.91	15.78	18.84	21.87	24.26
User CPU%	5.09	7.21	9.76	11.89	14.53	17.34	20.10	22.23
System CPU%	0.47	0.58	0.87	1.01	1.25	1.49	1.76	2.03
Hits/Sec	58.80	82.21	105.86	130.36	153.40	177.00	201.38	224.57
Avg Response	0.04	0.04	0.04	0.05	0.05	0.05	0.05	0.05
Users/CPU Core (@100%)	1124	1122	1058	1065	1030	995	972	979
Users/CPU Core (actual)	63	88	113	138	163	188	213	238
RAM/User (MB)	7.08	6.48	6.13	5.92	5.78	5.70	5.62	5.55

Users	2100	2300	2500	2700	2900	3100	3300	3500
No. of CPUs	4	4	4	4	4	4	4	4
No. of CPU Cores	8	8	8	8	8	8	8	8
Total CPU%	27.48	30.27	35.21	38.29	42.45	46.74	50.49	56.37
User CPU%	25.20	27.66	32.23	34.86	38.69	42.23	45.82	50.76
System CPU%	2.28	2.61	2.98	3.43	3.76	4.51	4.67	5.60
Hits/Sec	249.29	272.41	297.30	319.78	343.30	367.04	391.23	415.87
Avg Response	0.05	0.05	0.06	0.06	0.06	0.06	0.06	0.07
Users/CPU Core (@100%)	955	950	888	881	854	829	817	776
Users/CPU Core (actual)	263	288	313	338	363	388	413	438
RAM/User (MB)	5.50	5.46	5.44	5.42	5.37	5.37	5.36	5.34

Users	3700	3900	4100	4300	4500
No. of CPUs	4	4	4	4	4
No. of CPU Cores	8	8	8	8	8
Total CPU%	59.92	67.28	73.20	80.64	86.61
User CPU%	53.77	60.51	66.00	72.33	77.62
System CPU%	6.15	6.77	7.20	8.30	8.98
Hits/Sec	438.67	461.90	486.22	509.55	531.92
Avg Response	0.07	0.07	0.07	0.08	0.10
Users/CPU Core (@100%)	772	725	700	667	649
Users/CPU Core (actual)	463	488	513	538	563
RAM/User (MB)	5.34	5.33	5.27	5.27	5.20

Figure 15: x366 system 4-Way (8x 3.0 GHz Cores), 2MB L2

x460 System 8-Way (8x 3.3 GHz Cores), 8MB L3

Users	500	700	900	1100	1300	1500	1700	1900
No. of CPUs	8	8	8	8	8	8	8	8
No. of CPU Cores	8	8	8	8	8	8	8	8
Total CPU%	5.54	8.06	9.97	12.26	16.19	17.22	21.58	23.98
User CPU%	4.89	7.16	8.84	10.87	14.45	15.15	19.18	21.22
System CPU%	0.65	0.90	1.14	1.38	1.74	2.07	2.40	2.75
Hits/Sec	58.71	82.66	106.44	130.60	153.90	177.49	200.77	224.36
Avg Response	0.04	0.04	0.04	0.04	0.04	0.04	0.04	0.05
Users/CPU Core (@100%)	1129	1085	1128	1122	1003	1089	985	991
Users/CPU Core (actual)	63	88	113	138	163	188	213	238
RAM/User (MB)	6.97	6.40	6.09	5.87	5.74	5.67	5.59	5.51

Users	2100	2300	2500	2700	2900	3100	3300	3500
No. of CPUs	8	8	8	8	8	8	8	8
No. of CPU Cores	8	8	8	8	8	8	8	8
Total CPU%	27.63	30.47	34.28	38.56	43.40	48.37	53.73	60.42
User CPU%	24.34	26.72	30.09	33.89	37.92	42.28	46.67	52.59
System CPU%	3.28	3.75	4.18	4.67	5.48	6.09	7.05	7.83
Hits/Sec	248.78	272.91	296.02	320.93	343.96	367.29	391.96	415.07
Avg Response	0.05	0.05	0.05	0.05	0.05	0.06	0.06	0.06
Users/CPU Core (@100%)	950	943	912	875	835	801	768	724
Users/CPU Core (actual)	263	288	313	338	363	388	413	438
RAM/User (MB)	5.47	5.42	5.40	5.38	5.37	5.34	5.35	5.33

Users	3700	3900	4100	4300	4500	4700
No. of CPUs	8	8	8	8	8	8
No. of CPU Cores	8	8	8	8	8	8
Total CPU%	66.84	74.03	79.82	86.88	91.15	98.06
User CPU%	58.38	64.09	69.45	75.50	78.89	84.76
System CPU%	8.46	9.93	10.37	11.38	12.25	13.30
Hits/Sec	439.39	462.78	486.06	509.19	532.36	552.84
Avg Response	0.07	0.07	0.08	0.09	0.10	0.19
Users/CPU Core (@100%)	692	659	642	619	617	599
Users/CPU Core (actual)	463	488	513	538	563	588
RAM/User (MB)	5.31	5.30	5.29	5.29	5.27	5.26

Figure 16: x460 system 8-Way (8x 3.3 GHz Cores), 8MB L3

Combined results: Users versus total CPU percentage

Below, you will find the combined results of this benchmark for users, as compared to the total CPU usage.

Key to Abbreviations

x460 8-way = x460 8-way (8x 3.3 GHz Cores), 8MB L3
x366 8-way = x366 4-way (8x 3.0 GHz Cores), 2MB L2
x366 4-way = x366 4-way (4x 3.6 GHz Cores), 1MB L2
4-way #1 = Intel P-IV 4-way (4x 2.8 GHz Cores), 4MB L3
4-way #2 = Intel P-IV 4-way (4x 2.8 GHz Cores), 2MB L3
4-way #3 = Intel P-IV 4-way (4x 2.8 GHz Cores), 2MB L3 (HT OFF)

Table Data

	Users	400	500	600	700	800	900	1000
x460 8-way	Total CPU%		5.54		8.06		9.97	
x366 8-way	Total CPU%		5.56		7.80		10.64	
x366 4-way	Total CPU%		9.70		13.76		17.99	
4-way #1	Total CPU%		16.70	20.41	24.53	29.27	34.38	40.03
4-way #2	Total CPU%		18.42	22.77	27.77	33.00	38.60	45.93
4-way #3	Total CPU%	25.83	32.68	39.64	46.70	54.30	62.16	69.50

	Users	1100	1200	1300	1400	1500	1700	1900
x460 8-way	Total CPU%	12.26		16.19		17.22	21.58	23.98
x366 8-way	Total CPU%	12.91		15.78		18.84	21.87	24.26
x366 4-way	Total CPU%	25.03		29.72		37.21	42.23	51.91
4-way #1	Total CPU%	45.99	52.54	58.96	66.67	74.61		
4-way #2	Total CPU%	53.55	62.17	70.47	81.22	93.23		
4-way #3	Total CPU%	77.85	86.20	93.03				

	Users	2100	2300	2500	2700	2800	2900	3100
x460 8-way	Total CPU%	27.63	30.47	34.28	38.56		43.40	48.37
x366 8-way	Total CPU%	27.48	30.27	35.21	38.29		42.45	46.74
x366 4-way	Total CPU%	58.11	70.71	78.83	89.74	95.10		

	Users	3300	3500	3700	3900	4100	4300	4500
x460 8-way	Total CPU%	53.73	60.42	66.84	74.03	79.82	86.88	91.15
x366 8-way	Total CPU%	50.49	56.37	59.92	67.28	73.20	80.64	86.61
x366 4-way	Total CPU%							

	Users	4700
x460 8-way	Total CPU%	98.06

Figure 17: Table data for users versus total CPU percentage

Graphical data

The data shown in tabular form in Figure 17 is represented here graphically in Figure 18.

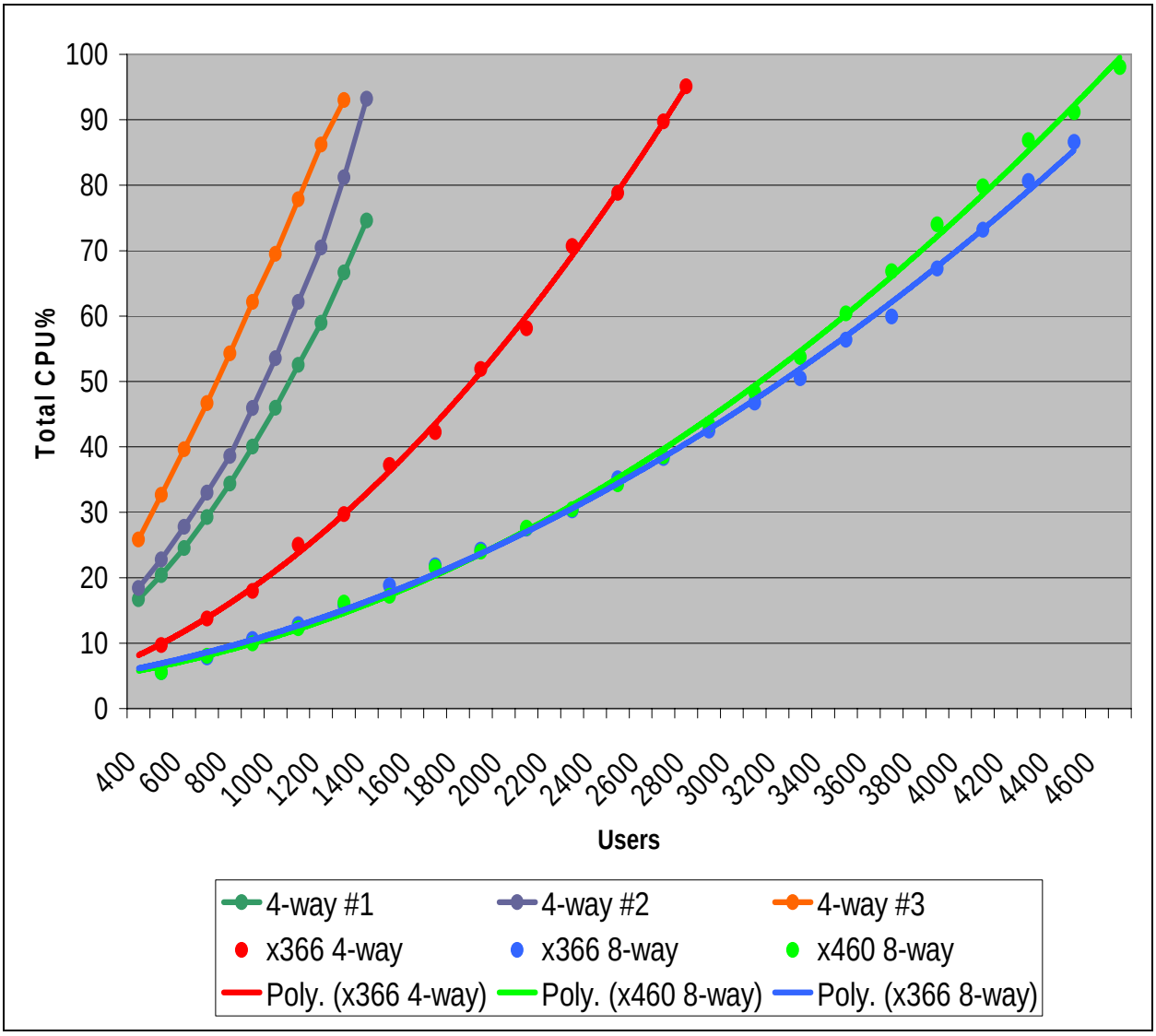


Figure 18: Users versus total CPU percentage

Combined Results: Total CPU percentage versus Users per CPU Core (actual)

The data in Figure 19 indicates the combined results of CPU usage versus actual users per CPU Core.

Key to Abbreviations

x460 8-way = x460 8-way (8x 3.3 GHz Cores), 8MB L3
x366 8-way = x366 4-way (8x 3.0 GHz Cores), 2MB L2
x366 4-way = x366 4-way (4x 3.6 GHz Cores), 1MB L2
4-way #1 = Intel P-IV 4-way (4x 2.8 GHz Cores), 4MB L3
4-way #2 = Intel P-IV 4-way (4x 2.8 GHz Cores), 2MB L3
4-way #3 = Intel P-IV 4-way (4x 2.8 GHz Cores), 2MB L3 (HT OFF)

Table Data

x460 8-way	Users/CPU Core (actual)	63	88	113	138	163	188	213	238
x460 8-way	Total CPU%	5.54	8.06	9.97	12.26	16.19	17.22	21.58	23.98
x366 8-way	Users/CPU Core (actual)	63	88	113	138	163	188	213	238
x366 8-way	Total CPU%	5.56	7.80	10.64	12.91	15.78	18.84	21.87	24.26
x366 4-way	Users/CPU Core (actual)	125	175	225	275	325	375	425	475
x366 4-way	Total CPU%	9.70	13.76	17.99	25.03	29.72	37.21	42.23	51.91
4-way #1	Users/CPU Core (actual)	125	150	175	200	225	250	275	300
4-way #1	Total CPU%	16.70	20.41	24.53	29.27	34.38	40.03	45.99	52.54
4-way #2	Users/CPU Core (actual)	125	150	175	200	225	250	275	300
4-way #2	Total CPU%	18.42	22.77	27.77	33.00	38.60	45.93	53.55	62.17
4-way #3	Users/CPU Core (actual)	100	125	150	175	200	225	250	275
4-way #3	Total CPU%	25.83	32.68	39.64	46.70	54.30	62.16	69.50	77.85

Figure 19: Table Data for Total CPU percentage versus Users per CPU Core (actual) (continued on next page)

x460 8-way	Users/CPU Core (actual)	263	288	313	338	363	388	413	438
x460 8-way	Total CPU%	27.63	30.47	34.28	38.56	43.40	48.37	53.73	60.42
x366 8-way	Users/CPU Core (actual)	263	288	313	338	363	388	413	438
x366 8-way	Total CPU%	27.48	30.27	35.21	38.29	42.45	46.74	50.49	56.37
x366 4-way	Users/CPU Core (actual)	525	575	625	675	700			
x366 4-way	Total CPU%	58.11	70.71	78.83	89.74	95.10			
4-way #1	Users/CPU Core (actual)	325	350	375					
4-way #1	Total CPU%	58.96	66.67	74.61					
4-way #2	Users/CPU Core (actual)	325	350	375					
4-way #2	Total CPU%	70.47	81.22	93.23					
4-way #3	Users/CPU Core (actual)	300	325						
4-way #3	Total CPU%	86.20	93.03						

x460 8-way	Users/CPU Core (actual)	463	488	513	538	563	588
x460 8-way	Total CPU%	66.84	74.03	79.82	86.88	91.15	98.06
x366 8-way	Users/CPU Core (actual)	463	488	513	538	563	
x366 8-way	Total CPU%	59.92	67.28	73.20	80.64	86.61	

Figure 20: Table Data for Total CPU percentage versus Users per CPU Core (actual) (continued from previous page)

Graphical data

The data shown in tabular form in Figures 19 and 20 is represented here graphically in Figure 21.

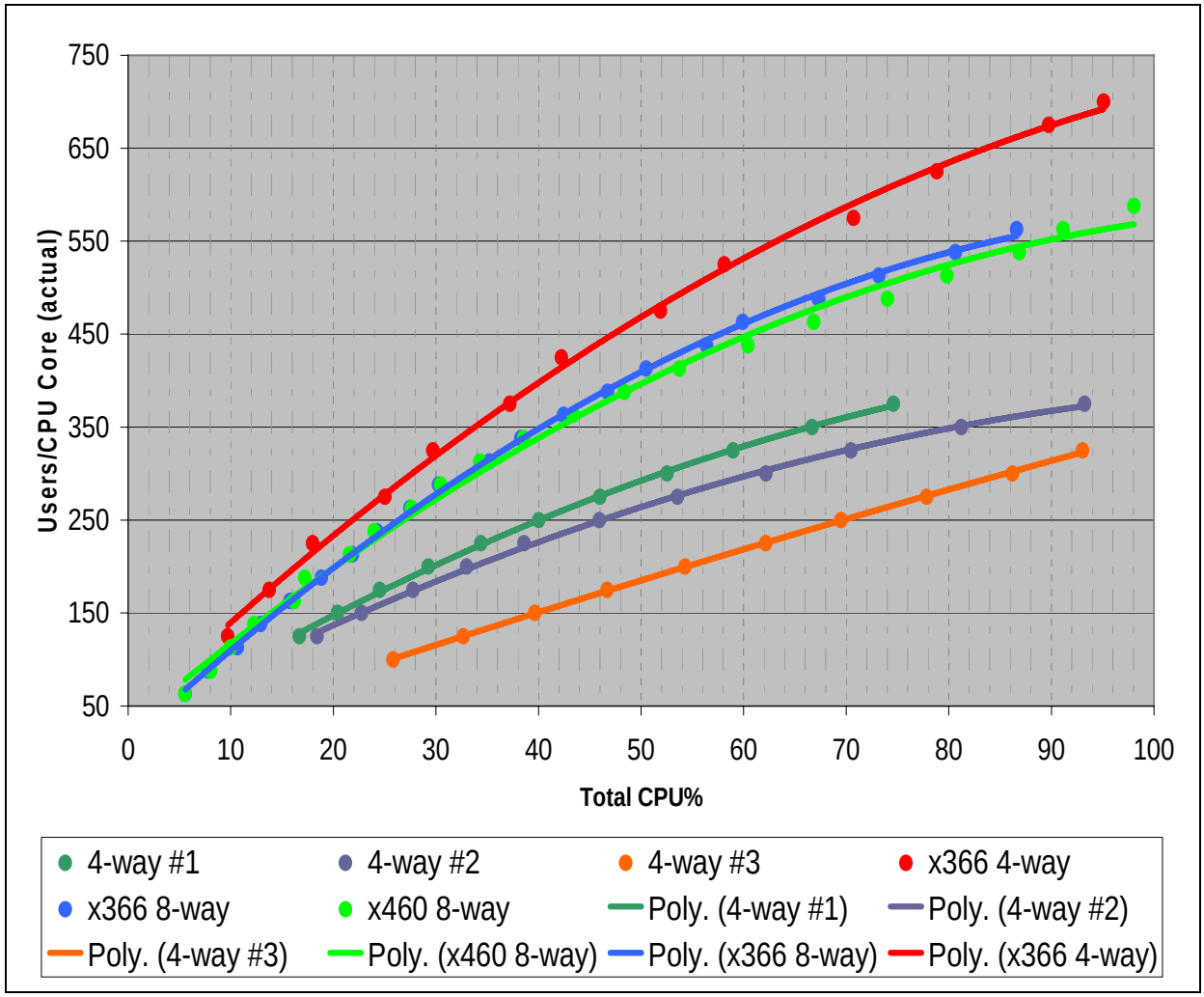


Figure 21: Graphical Data for Total CPU percentage versus Users per CPU Core (actual)

Trend line equations from graphical data

The **Total CPU percentage versus Users per CPU Core (actual)** graph in Figure 21 gives the testing team a way of predicting how many Siebel 7.7 users might be supported per CPU Core, given a desired CPU utilization level. Using the **polynomial trend lines** on the graph, the testing team can derive an equation for each type of server, as follows:

- **IBM x460 system 8-way (8x 3.3-GHz Cores), 8-MB L3**
  - Users/CPU Core = $-0.0388(\text{CPU\%})^2 + 9.3179(\text{CPU\%}) + 27.608$
    - @ **80%** CPU, Users/CPU Core = **525**
- **IBM x366 system 8-way (8x 3.0-GHz Cores), 2-MB L2**
  - Users/CPU Core = $-0.0456(\text{CPU\%})^2 + 10.226(\text{CPU\%}) + 12.076$
    - @ **80%** CPU, Users/CPU Core = **538**
- **IBM x366 system 4-way (4x 3.6-GHz Cores), 1-MB L2**
  - Users/CPU Core = $-0.0383(\text{CPU\%})^2 + 10.516(\text{CPU\%}) + 38.297$
    - @ **80%** CPU, Users/CPU Core = **634**
- **Intel P-IV 4-way (4x 2.8-GHz Cores), 4-MB L3**
  - Users/CPU Core = $-0.0283(\text{CPU\%})^2 + 6.8137(\text{CPU\%}) + 22.583$
    - @ **80%** CPU, Users/CPU Core = **387**
- **Intel P-IV 4-way (4x 2.8-GHz Cores), 2-MB L3**
  - Users/CPU Core = $-0.0236(\text{CPU\%})^2 + 5.8873(\text{CPU\%}) + 28.52$
    - @ **80%** CPU, Users/CPU Core = **348**
- **Intel P-IV 4-way (4x 2.8-GHz Cores), 2-MB L3 (HT OFF)**
  - Users/CPU Core = $-0.0043(\text{CPU\%})^2 + 3.8124(\text{CPU\%}) + 5.0999$
    - @ **80%** CPU, Users/CPU Core = **283**

Conclusions

The graphs clearly demonstrate that under the same Siebel 7.7 testing conditions, the IBM third generation X3 Architecture-based x366 and x460 systems offer dramatically greater scalability than the Intel 4-way white-box servers do.

The best results from the Siebel study came from the **Intel P-IV 4-way (4x 2.8 GHz Cores), 4MB L3** server. At 80% CPU utilization, you can expect to support about 387 Siebel 7.7 users per CPU Core. The best results from the IBM study, in terms of users per CPU Core, came from the **IBM x366 system 4-way (4x 3.6 GHz Cores), 1MB L2** server. At 80% CPU utilization, you can see a dramatic jump to 634 Siebel 7.7 users per CPU Core. This is a 64% increase in performance over the best Intel 4-way white-box server.

The xSeries X3 Architecture servers look even more impressive if you look at them in terms of the total number of Siebel 7.7 users that can be supported. Again, the best results from the Siebel study came from the **Intel P-IV 4-way (4x 2.8 GHz Cores), 4MB L3** server. It achieved a maximum of 1,500 Siebel 7.7 users at 75% CPU utilization.

This time, the best result from the IBM study came from the **IBM x366 system 8-way (8x 3.0 GHz Cores), 2MB L2** server, which achieved a maximum of 4,500 Siebel 7.7 users at 87% CPU utilization. This is a 200% increase in Siebel 7.7 users over the best Intel 4-way white-box server.

Another perspective on the performance of the X3 Architecture can be obtained by focusing on the best 4-way server results. The best results obtained in this category were as follows:

- **IBM x366 system 4-way (4x 3.6 GHz Cores), 1MB L2**
  - 634 Users per CPU Core at 80% CPU utilization
- **Intel P-IV 4-way (4x 2.8 GHz Cores), 4MB L3**
  - 387 Users per CPU Core at 80% CPU utilization

Looking at the CPU clock speeds, you can see that the x366 system has a 29% increase in clock speed over the Intel white-box server; that is, (((3.6 – 2.8) / 2.8) * 100) = 29%. Looking at the users per CPU Core achieved, the x366 system supported 64% more users than the Intel server; that is, (((634 – 387) / 387) * 100) = 64%. The x366 system exhibited about 2.2x more performance than expected based on CPU clock speed alone. This significant performance boost is provided by the X3 technology.

It is also clear from the results that the X3 Architecture is not just dual-core CPU capable, it actually excels when fitted with them. Looking back at the Siebel 7.7 users per CPU Core results, you can see that the **IBM x366 system 8-way (8x 3.0 GHz Cores), 2MB L2** server (that is, the x366 system fitted with dual-core CPUs) achieved 538 Siebel 7.7 users per CPU Core at 80% CPU utilization. The **IBM x460 system 8-way (8x 3.3 GHz Cores), 8MB L3** [3] server (that is, the x460 system fitted with single-core CPUs in a two-chassis, single-partition configuration) achieved 525 Siebel 7.7 users per CPU Core at 80% CPU. The dual-core CPUs showed a 2.5% increase in performance over the equivalent number of single-core CPU cores.

However, with a little extra physical RAM, the x366 system fitted with dual-core CPUs might only achieve a couple hundred more Siebel 7.7 users before maxing out completely. The x460 system has the advantage here, in that extra chassis can be added to the partition, up to a maximum of eight chassis. At the time this paper was written, it was unknown how Siebel application software might scale in such a large partition.

Also note that the x460 system is itself, dual-core CPU capable. An interesting follow-on study would be to see how Siebel 7.7 application software might scale on a two-chassis x460 system, fitted with dual-core CPUs and a full complement of 64GB of RAM.

[3] The **IBM x460 system 8-way (8x 3.3 GHz Cores), 8MB L3** server actually achieved a higher Siebel 7 user count of 4,700 users. However, this is only because the x366 system fitted with the dual-core CPUs ran out of RAM. The trend lines on the graph clearly show that the **IBM x366 system 8-way (8x 3.0 GHz Cores), 2MB L2** server would have achieved more physical Siebel 7 users if more physical RAM had been available.

PSPP benchmark

A total of four significant x366 and x460 machine configurations were assembled for the PSPP benchmark runs:

- x366 system with 4x single-core 3.6 GHz Intel Xeon CPUs (4x 3.6 GHz CPU Cores)
- x366 system with 4x dual-core 3.0 GHz Intel Xeon CPUs (8x 3.0 GHz CPU Cores)
- x460 system with 4x single-core 3.3 GHz Intel Xeon CPUs (4x 3.3 GHz CPU Cores)
- x460 system with 8x single-core 3.3 GHz Intel Xeon CPUs (8x 3.3 GHz CPU Cores)

This allowed the testing team to compare the performance of Siebel 7.7 in 4-way and 8-way CPU Core configurations for both xSeries systems. The following sections present the main PSPP benchmark data that was captured for each run.

Siebel Application Server

The x366 and x460 systems were running the Siebel 7.7 Application Server supporting the front-end client workload. The Siebel Gateway was also installed on these servers, but the Gateway was idle during a run and generated no system load.

The statistics reported in Figures 22 through 25 are color-coded to allow easier identification of each result set on the graphs in Figures 42 and 43 that follow later in this section.

x366 System single-core PSPP

Figure 22 shows the statistics captured for the x366 system fitted with 4x single-core, 3.6 GHz Intel Xeon MP CPUs. This was a single-chassis configuration which had 4x 3.6 GHz CPU Cores.

Users	No. of CPUs	No. of CPU Cores	Steady State Total CPU (%)	Users/CPU Core (@100%)	Users/CPU Core (actual)	Steady State Free RAM (MB)	Total Installed RAM (MB)	RAM Used (MB)	RAM/User (MB)
2,100	4	4	83.4	629	525	12,768	24,576	11,808	5.62

Figure 22: x366 system single-core PSPP: Siebel Application Server Statistics

x366 System Dual-Core PSPP

Figure 23 shows the statistics captured for the x366 system fitted with 4x dual-core, 3.0 GHz Intel Xeon MP CPUs. This was a single-chassis configuration which had 8x 3.0 GHz CPU Cores.

Users	No. of CPUs	No. of CPU Cores	Steady State Total CPU (%)	Users/CPU Core (@100%)	Users/CPU Core (actual)	Steady State Free RAM (MB)	Total Installed RAM (MB)	RAM Used (MB)	RAM/User (MB)
3,600	4	8	84.7	531	450	5,158	24,576	19,418	5.39

Figure 23: x366 system dual-core PSPP: Siebel Application Server Statistics

x460 System single-core, single-chassis PSPP

Figure 24 shows the statistics captured for the x460 system fitted with 4x single-core, 3.3 GHz Intel Xeon MP CPUs. This was a single-chassis configuration which had 4x 3.3 GHz CPU Cores (each chassis has 4x single-core CPUs).

Users	No. of CPUs	No. of CPU Cores	Steady State Total CPU (%)	Users/CPU Core (@100%)	Users/CPU Core (actual)	Steady State Free RAM (MB)	Total Installed RAM (MB)	RAM Used (MB)	RAM/User (MB)
2,000	4	4	84.0	596	500	5,192	16,384	11,192	5.60

Figure 24: x460 system Single-Chassis PSPP: Siebel Application Server Statistics

x460 System single-core, two-chassis PSPP

Figure 25 shows the statistics captured for the x460 system fitted with 8x single-core, 3.3 GHz Intel Xeon MP CPUs. This was a two-chassis, single-partition configuration which had 8x 3.3 GHz CPU Cores (each chassis has 4x single-core CPUs).

Users	No. of CPUs	No. of CPU Cores	Steady State Total CPU (%)	Users/CPU Core (@100%)	Users/CPU Core (actual)	Steady State Free RAM (MB)	Total Installed RAM (MB)	RAM Used (MB)	RAM/User (MB)
3,400	8	8	83.8	507	425	13,612	32,256	18,644	5.48

Figure 25: x460 system two-chassis PSPP: Siebel Application Server Statistics

Batch HTTP workload

The following sections show the statistics that were collected on the BladeCenter HS20 system that was driving the Siebel 7.7 batch workload. The workload was generated by six worker threads; each running a Java-based HTTP script. The threads were running while the Siebel 7.7 client application server was in steady state.

The statistics for each thread are quoted in average operations per second (ops). These ops are accumulated and converted to average operations per hour (oph). Siebel mandates targets for oph that must be achieved for the PSPP results to be valid. The oph statistics quoted in Figure 26 through Figure 29 were well above that required by Siebel for a 5,000 user PSPP run. Accordingly, these results were valid for the PSPP runs.

x366 System single-core PSPP

```
HTTP KPIS: Generated by scriptHTTP.pl

Worker_0    60000    Nov 28, 2005 9:58:23 PM    Nov 28, 2005 11:06:33 PM    17.24
Worker_1    60000    Nov 28, 2005 9:58:25 PM    Nov 28, 2005 11:07:37 PM    16.93
Worker_2    60000    Nov 28, 2005 9:58:27 PM    Nov 28, 2005 11:07:10 PM    17.05
Worker_3    60000    Nov 28, 2005 9:58:29 PM    Nov 28, 2005 11:06:51 PM    17.15
Worker_4    60000    Nov 28, 2005 9:58:31 PM    Nov 28, 2005 11:07:37 PM    16.93
Worker_5    60000    Nov 28, 2005 9:58:33 PM    Nov 28, 2005 11:07:00 PM    17.11
Avg thruput:   102.41         (ops)
Avg thruput:   368658.92      (oph)   Target: (5kclients) 120K
```

Figure 26: x366 System single-core PSPP: HTTP Batch Workload Statistics

x366 System dual-core PSPP

```
HTTP KPIS: Generated by scriptHTTP.pl

Worker_0    60000    Nov 23, 2005 1:44:40 PM    Nov 23, 2005 2:57:48 PM       15.86
Worker_1    60000    Nov 23, 2005 1:44:42 PM    Nov 23, 2005 2:58:29 PM       15.69
Worker_2    60000    Nov 23, 2005 1:44:44 PM    Nov 23, 2005 2:57:49 PM       15.86
Worker_3    60000    Nov 23, 2005 1:44:46 PM    Nov 23, 2005 2:57:48 PM       15.87
Worker_4    60000    Nov 23, 2005 1:44:48 PM    Nov 23, 2005 2:57:50 PM       15.87
Worker_5    60000    Nov 23, 2005 1:44:50 PM    Nov 23, 2005 2:58:12 PM       15.78
Avg thruput:   94.93          (ops)
Avg thruput:   341756.19      (oph)   Target: (5kclients) 120K
```

Figure 27: x366 System dual-core PSPP: HTTP Batch Workload Statistics

x460 System single-core, single-chassis PSPP

```
HTTP KPIS: Generated by scriptHTTP.pl

Worker_0    60000    Dec 7, 2005 2:25:38 PM    Dec 7, 2005 3:33:58 PM    17.17
Worker_1    60000    Dec 7, 2005 2:25:40 PM    Dec 7, 2005 3:33:33 PM    17.29
Worker_2    60000    Dec 7, 2005 2:25:42 PM    Dec 7, 2005 3:33:29 PM    17.31
Worker_3    60000    Dec 7, 2005 2:25:44 PM    Dec 7, 2005 3:34:46 PM    16.94
Worker_4    60000    Dec 7, 2005 2:25:46 PM    Dec 7, 2005 3:34:39 PM    16.98
Worker_5    60000    Dec 7, 2005 2:25:48 PM    Dec 7, 2005 3:33:55 PM    17.21
Avg thruput:   102.91         (ops)
Avg thruput:   370471.56      (oph)   Target: (5kclients) 120K
```

Figure 28: x460 System Single-Chassis PSPP: HTTP Batch Workload Statistics

x460 System single-core, two-chassis PSPP

```
HTTP KPIS: Generated by scriptHTTP.pl

Worker_0    60000    Dec 6, 2005 1:35:37 PM    Dec 6, 2005 2:48:14 PM    16.00
Worker_1    60000    Dec 6, 2005 1:35:39 PM    Dec 6, 2005 2:48:26 PM    15.95
Worker_2    60000    Dec 6, 2005 1:35:41 PM    Dec 6, 2005 2:47:48 PM    16.11
Worker_3    60000    Dec 6, 2005 1:35:43 PM    Dec 6, 2005 2:48:56 PM    15.82
Worker_4    60000    Dec 6, 2005 1:35:45 PM    Dec 6, 2005 2:47:51 PM    16.11
Worker_5    60000    Dec 6, 2005 1:35:47 PM    Dec 6, 2005 2:47:42 PM    16.15
Avg thruput:   96.13          (ops)
Avg thruput:   346052.94      (oph)   Target: (5kclients) 120K
```

Figure 29: x460 System two-chassis PSPP: HTTP Batch Workload Statistics

Database server

The pSeries p650 system used for the database server was reconfigured as a 4-way machine for all of the PSPP runs. This was because the load generated by the x366 and x460 systems was not enough to stress significantly the p650 system with eight CPUs.

The database server statistics reported below were captured during the steady states of the PSPP runs, when the batch and client workloads were both running.

Before each PSPP run, an SQL query was run to determine the row counts of every table in the Siebel PSPP database schema. The same query was executed after each run. In the database table row statistics quoted in Figures 31, 33, 35 and 37, only those Siebel database tables that exhibited some growth have been reported. All other database tables were not updated by the PSPP runs, although some of them might have been read.

x366 System single-core PSPP

Machine	TotCPU	UsrCPU	SysCPU	WaiCPU	ConSwitc	Syscall	Procs	Threads	MemUsed
p650	38.68	33.51	5.17	6.88	4990	13183	2202	2202	8650928

Figure 30: x366 System single-core PSPP: Database server statistics

Table	Before	After	Growth
SIEBEL.S_DOCK_TXN_LOG	0	1129973	1129973
SIEBEL.S_EVT_ACT	6685419	6832209	146790
SIEBEL.S_SRV_ACT	37995	178772	140777
SIEBEL.S_ACT_EMP	2233637	2355911	122274
SIEBEL.S_ACT_CONTACT	3371806	3488067	116261
SIEBEL.S_SRV_REQ	5651814	5694527	42713
SIEBEL.S_SRV_REQ1_FNX	56851	99564	42713
SIEBEL.S_SRV_REQ2_FNX	56851	99564	42713
SIEBEL.S_SRV_REQ3_FNX	56865	99578	42713
SIEBEL.S_SRV_REQ_BU	5651770	5694483	42713
SIEBEL.S_SRV_REQ_LOYX	56766	99479	42713
SIEBEL.S_SUSP_ACT	714	25230	24516
SIEBEL.S_EVT_ACT_FNX	87448	105645	18197
SIEBEL.S_EVT_CAL	35670	53867	18197
SIEBEL.S_EVT_MKTG	35670	53867	18197
SIEBEL.S_OPTY	3429952	3448149	18197
SIEBEL.S_OPTY_BU	3317919	3336116	18197
SIEBEL.S_OPTY_POSTN	3331541	3349738	18197
SIEBEL.S_OPTY_UTX	35119	53316	18197
SIEBEL.S_REVN	2954675	2972872	18197
SIEBEL.S_SALES_ACT	36047	54244	18197
SIEBEL.S_OPTY_DSGN_REG	35126	53323	18197
SIEBEL.S_OPTY_TNTX	35122	53319	18197

Figure 31: x366 System single-core PSPP: Database table row statistics

x366 System dual-core PSPP

Machine	TotCPU	UsrCPU	SysCPU	WaiCPU	ConSwitc	Syscall	Procs	Threads	MemUsed
p650	69.87	61.29	8.58	5.64	6651	20117	3702	3702	14665440

Figure 32: x366 System dual-core PSPP: Database server statistics

Table	Before	After	Growth
SIEBEL.S_DOCK_TXN_LOG	0	1988750	1988750
SIEBEL.S_EVT_ACT	6685419	6947528	262109
SIEBEL.S_SRV_ACT	37995	294081	256086
SIEBEL.S_ACT_EMP	2233637	2451080	217443
SIEBEL.S_ACT_CONTACT	3371806	3583226	211420
SIEBEL.S_SRV_REQ	5651814	5729236	77422
SIEBEL.S_SRV_REQ1_FNX	56851	134273	77422
SIEBEL.S_SRV_REQ2_FNX	56851	134273	77422
SIEBEL.S_SRV_REQ3_FNX	56865	134287	77422
SIEBEL.S_SRV_REQ_BU	5651770	5729192	77422
SIEBEL.S_SRV_REQ_LOYX	56766	134188	77422
SIEBEL.S_SUSP_ACT	714	45380	44666
SIEBEL.S_EVT_ACT_FNX	87448	120204	32756
SIEBEL.S_EVT_CAL	35670	68426	32756
SIEBEL.S_EVT_MKTG	35670	68426	32756
SIEBEL.S_OPTY	3429952	3462708	32756
SIEBEL.S_OPTY_BU	3317919	3350675	32756
SIEBEL.S_OPTY_POSTN	3331541	3364297	32756
SIEBEL.S_OPTY_UTX	35119	67875	32756
SIEBEL.S_REVN	2954675	2987431	32756
SIEBEL.S_SALES_ACT	36047	68803	32756
SIEBEL.S_OPTY_DSGN_REG	35126	67882	32756
SIEBEL.S_OPTY_TNTX	35122	67878	32756

Figure 33: x366 System dual-core PSPP: Database table row statistics

x460 System single-core, single-chassis PSPP

Machine	TotCPU	UsrCPU	SysCPU	WaiCPU	ConSwitc	Syscall	Procs	Threads	MemUsed
p650	37.45	32.51	4.94	7.36	4860	12955	2089	2089	8223336

Figure 34: x460 System single-chassis PSPP: Database server statistics

Table	Before	After	Growth
SIEBEL.S_DOCK_TXN_LOG	472468	1736914	1264446
SIEBEL.S_EVT_ACT	6748921	6914187	165266
SIEBEL.S_SRV_ACT	101497	260751	159254
SIEBEL.S_ACT_EMP	2286037	2423523	137486
SIEBEL.S_ACT_CONTACT	3424206	3555680	131474
SIEBEL.S_SRV_REQ	5670908	5719042	48134
SIEBEL.S_SRV_REQ1_FNX	75945	124079	48134
SIEBEL.S_SRV_REQ2_FNX	75945	124079	48134
SIEBEL.S_SRV_REQ3_FNX	75959	124093	48134
SIEBEL.S_SRV_REQ_BU	5670864	5718998	48134
SIEBEL.S_SRV_REQ_LOYX	75860	123994	48134
SIEBEL.S_SUSP_ACT	11816	39596	27780
SIEBEL.S_EVT_ACT_FNX	95440	115794	20354
SIEBEL.S_EVT_CAL	43662	64016	20354
SIEBEL.S_EVT_MKTG	43662	64016	20354
SIEBEL.S_OPTY	3437944	3458298	20354
SIEBEL.S_OPTY_BU	3325911	3346265	20354
SIEBEL.S_OPTY_POSTN	3339533	3359887	20354
SIEBEL.S_OPTY_UTX	43111	63465	20354
SIEBEL.S_REVN	2962667	2983021	20354
SIEBEL.S_SALES_ACT	44039	64393	20354
SIEBEL.S_OPTY_DSGN_REG	43118	63472	20354
SIEBEL.S_OPTY_TNTX	43114	63468	20354

Figure 35: x460 System single-chassis PSPP: Database table row statistics

x460 System single-core, two-chassis PSPP

Machine	TotCPU	UsrCPU	SysCPU	WaiCPU	ConSwitc	Syscall	Procs	Threads	MemUsed
p650	65.35	57.27	8.08	6.51	6498	19068	3506	3506	13835012

Figure 36: x460 System two-chassis PSPP: Database server statistics

Table	Before	After	Growth
SIEBEL.S_DOCK_TXN_LOG	0	1933104	1933104
SIEBEL.S_EVT_ACT	6685419	6939810	254391
SIEBEL.S_SRV_ACT	37995	286377	248382
SIEBEL.S_ACT_EMP	2233637	2444728	211091
SIEBEL.S_ACT_CONTACT	3371806	3576888	205082
SIEBEL.S_SRV_REQ	5651814	5726996	75182
SIEBEL.S_SRV_REQ1_FNX	56851	132033	75182
SIEBEL.S_SRV_REQ2_FNX	56851	132033	75182
SIEBEL.S_SRV_REQ3_FNX	56865	132047	75182
SIEBEL.S_SRV_REQ_BU	5651770	5726952	75182
SIEBEL.S_SRV_REQ_LOYX	56766	131948	75182
SIEBEL.S_SUSP_ACT	714	44014	43300
SIEBEL.S_EVT_ACT_FNX	87448	119330	31882
SIEBEL.S_EVT_CAL	35670	67552	31882
SIEBEL.S_EVT_MKTG	35670	67552	31882
SIEBEL.S_OPTY	3429952	3461834	31882
SIEBEL.S_OPTY_BU	3317919	3349801	31882
SIEBEL.S_OPTY_POSTN	3331541	3363423	31882
SIEBEL.S_OPTY_UTX	35119	67001	31882
SIEBEL.S_REVN	2954675	2986557	31882
SIEBEL.S_SALES_ACT	36047	67929	31882
SIEBEL.S_OPTY_DSGN_REG	35126	67008	31882
SIEBEL.S_OPTY_TNTX	35122	67004	31882

Figure 37: x460 System two-chassis PSPP: Database table row statistics

FastT storage server

A BladeCenter HS20 system was used to run FastT Storage Manager Software. The software monitored the DS4300 Storage Server and captured statistics for the PSPP runs.

The FastT statistics reported in Figures 38 through 41 were captured during the steady states of the PSPP runs, when the batch and client workloads were both running.

x366 System single-core PSPP

```
Performance Monitor data for Storage Subsystem SiebelNew1722

                        Total    Read  Cache Current Maximum Current Maximum
Devices                 IOs      %     Hit%  KB/sec  KB/sec  IO/sec  IO/sec
CONTROLLER IN SLOT B    0        0.0   0.0   0.0     0.0     0.0     0.0
CONTROLLER IN SLOT A    2009574  41.1  82.8  2470.6  3146.9  446.6   512.6
Logical Drive LD1       354114   15.9  36.4  606.4   783.1   77.5    93.2
Logical Drive LD2       579913   49.0  87.5  620.4   788.3   128.6   147.0
Logical Drive LD3       518962   43.1  84.0  624.0   792.3   116.3   132.5
Logical Drive LD4       556585   47.1  86.5  619.8   783.2   124.3   140.8
STORAGE SUBSYSTEM TOTALS 2009574 41.1  82.8  2470.6  3146.9  446.6   512.6

Start: 11/28/05 9:58:45 PM
Stop: 11/28/05 11:12:59 PM
Time Monitored: 01:14:14
Polling Interval (in seconds): 300
```

Figure 38: x366 System single-core PSPP: DS4300 FastT Storage Server statistics

x366 System dual-core PSPP

```
Performance Monitor data for Storage Subsystem SiebelNew1722

                        Total    Read  Cache Current Maximum Current Maximum
Devices                 IOs      %     Hit%  KB/sec  KB/sec  IO/sec  IO/sec
CONTROLLER IN SLOT B    0        0.0   0.0   0.0     0.0     0.0     0.0
CONTROLLER IN SLOT A    3193239  48.5  81.8  3473.5  4560.9  624.6   690.5
Logical Drive LD1       507107   18.5  23.4  840.4   1113.8  94.7    111.1
Logical Drive LD2       948794   56.9  86.9  885.2   1149.4  188.1   204.6
Logical Drive LD3       831487   50.5  83.3  874.4   1146.1  163.2   180.1
Logical Drive LD4       905851   54.6  86.1  873.4   1151.6  178.6   195.7
STORAGE SUBSYSTEM TOTALS 3193239 48.5  81.8  3473.5  4560.9  624.6   690.5

Start: 11/23/05 1:44:35 PM
Stop: 11/23/05 3:06:38 PM
Time Monitored: 01:22:03
Polling Interval (in seconds): 300
```

Figure 39: x366 System dual-core PSPP: DS4300 FastT Storage Server statistics

x460 System single-core, single-chassis PSPP

```
Performance Monitor data for Storage Subsystem SiebelNew1722

                        Total    Read  Cache Current Maximum Current Maximum
Devices                 IOs      %     Hit%  KB/sec  KB/sec  IO/sec  IO/sec
CONTROLLER IN SLOT B    0        0.0   0.0   0.0     0.0     0.0     0.0
CONTROLLER IN SLOT A    2552570  40.6  83.3  2071.0  3214.5  397.7   484.4
Logical Drive LD1       447210   14.8  32.1  504.4   785.1   65.4    86.8
Logical Drive LD2       739501   48.7  87.9  519.8   809.9   116.0   138.9
Logical Drive LD3       659903   42.5  84.8  527.0   805.1   104.9   125.8
Logical Drive LD4       705956   46.8  87.2  519.8   814.3   111.4   132.9
STORAGE SUBSYSTEM TOTALS 2552570 40.6  83.3  2071.0  3214.5  397.7   484.4

Start: 12/7/05 2:25:49 PM
Stop: 12/7/05 4:03:04 PM
Time Monitored: 01:37:15
Polling Interval (in seconds): 300
```

Figure 40: x460 System single-chassis PSPP: DS4300 FastT Storage Server statistics

x460 System single-core, two-chassis PSPP

```
Performance Monitor data for Storage Subsystem SiebelNew1722

                         Total    Read  Cache Current Maximum Current Maximum
Devices                  IOs      %     Hit%  KB/sec  KB/sec  IO/sec  IO/sec
CONTROLLER IN SLOT B     0        0.0   0.0   0.0     0.0     0.0     0.0
CONTROLLER IN SLOT A     3119031  47.6  81.2  3283.8  4725.6  606.7   693.4
Logical Drive LD1        502464   18.6  24.9  793.7   1161.8  93.8    116.7
Logical Drive LD2        921361   55.8  86.5  837.7   1185.6  181.8   201.3
Logical Drive LD3        812944   49.5  82.5  831.3   1187.2  158.5   180.7
Logical Drive LD4        882262   53.7  85.4  821.0   1191.0  172.6   194.7
STORAGE SUBSYSTEM TOTALS 3119031  47.6  81.2  3283.8  4725.6  606.7   693.4

Start: 12/6/05 1:35:51 PM
Stop: 12/6/05 2:57:16 PM
Time Monitored: 01:21:24
Polling Interval (in seconds): 300
```

Figure 41: x460 System two-chassis PSPP: DS4300 FastT Storage Server statistics

Graphical data

A lot of statistical data was captured during the PSPP benchmark. The most salient data to represent in graphical form is the number of Siebel 7.7 client users that can be supported per CPU Core. The graphs in Figures 42 and 43 represent two views of this particular statistic:

Users per CPU Core (@100%)

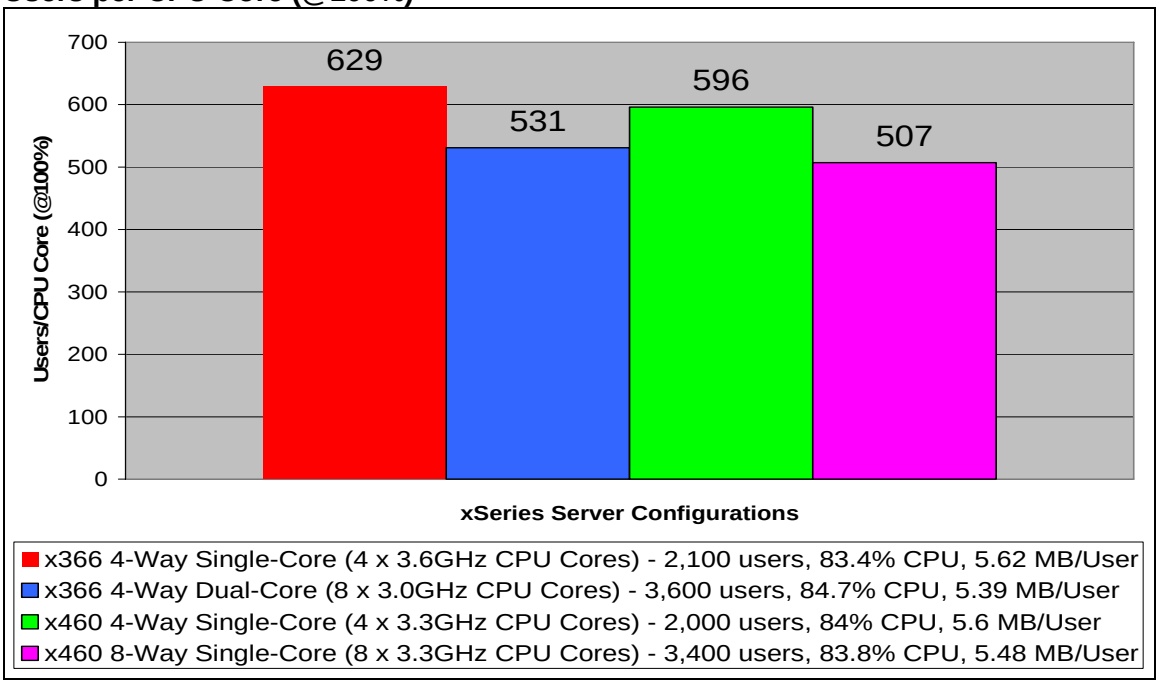


Figure 42: x366 and x460 System results combined

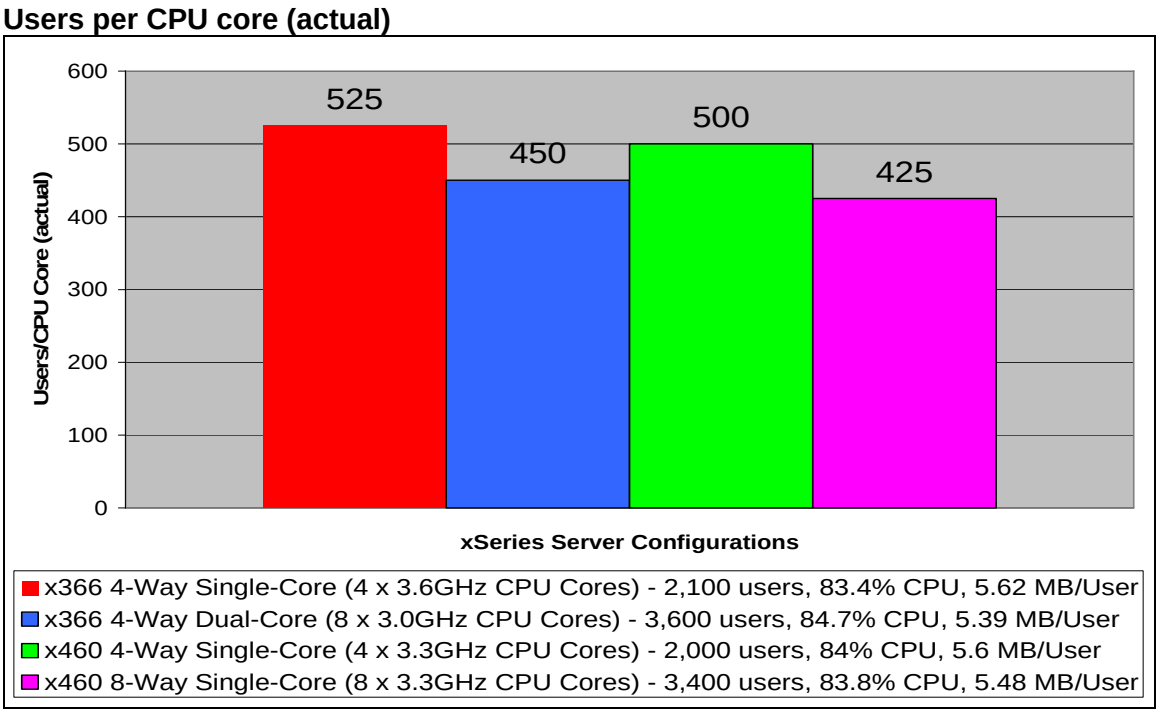


Figure 43: x366 and x460 System results combined (actual)

Conclusions

Many PSPP benchmarks have been performed with Siebel 7.7 application software on many different hardware configurations. The usual goal of a PSPP benchmark is to achieve a new high watermark for the total number of Siebel 7.7 users that can be run in an enterprise spread across many application servers and machines.

However, the PSPP benchmarks run on these x366 and x460 systems have a different objective. Here the objective was to see how high one instance of a Siebel Application Server can scale on each of the X3 Architecture-based xSeries systems.

No other PSPP benchmarks of this type exist that can be used for a direct comparison with the results published here. It might be possible to extrapolate results published for other PSPP benchmarks run on similar technology, but that analysis is very lengthy and beyond the scope of this paper.

Regardless, the results published here are impressive. The testing team achieved a high number of Siebel 7.7 users per CPU Core with the single-core 3.6 GHz CPUs producing the best result. However, the highest total number of Siebel 7.7 users was achieved with the x366 system fitted with dual-core 3.0 GHz CPUs, with the x460 system configured with 8x 3.3 GHz CPU Cores coming in a close second. The total user counts of 3,600 and 3,400 users respectively, are impressive totals for a single Siebel 7.7 Application Server running under Windows Server 2003.

Summary

This paper provided the reader with an introduction to the third-generation X3 Architecture xSeries systems. It summarized the main features that make the X3 Architecture better than the preceding technologies and then went on to show the performance improvements realized by the x366 and x460 systems with Siebel 7.7 Enterprise Software.

Please refer to **Appendix I**, for links to sites that contain additional information about the xSeries product line.

Appendix A: x366 System and Windows Server 2003 EE Configuration Information

Configuration data that appears here was captured using the Windows **msinfo32** utility.

Single-Core Intel Xeon MP CPUs

Item	Value
OS Name	Microsoft(R) Windows(R) Server 2003, Enterprise Edition
Version	5.2.3790 Service Pack 1 Build 3790
Other OS Description	Not Available
OS Manufacturer	Microsoft Corporation
System Name	EL9-92-154
System Manufacturer	IBM
System Model	eserver xSeries 366-[88632RU]-
System Type	X86-based PC
Processor	X86 Family 15 Model 4 Stepping 1 GenuineIntel ~3670 Mhz
Processor	X86 Family 15 Model 4 Stepping 1 GenuineIntel ~3670 Mhz
Processor	X86 Family 15 Model 4 Stepping 1 GenuineIntel ~3670 Mhz
Processor	X86 Family 15 Model 4 Stepping 1 GenuineIntel ~3670 Mhz
Processor	X86 Family 15 Model 4 Stepping 1 GenuineIntel ~3670 Mhz
Processor	X86 Family 15 Model 4 Stepping 1 GenuineIntel ~3670 Mhz
Processor	X86 Family 15 Model 4 Stepping 1 GenuineIntel ~3670 Mhz
Processor	X86 Family 15 Model 4 Stepping 1 GenuineIntel ~3670 Mhz
BIOS Version/Date	IBM -[ZUE136AUS-1.04]-, 6/28/2005
SMBIOS Version	2.3
Windows Directory	C:\WINDOWS
System Directory	C:\WINDOWS\system32
Boot Device	\Device\HarddiskVolume1
Locale	United States
Hardware Abstraction Layer	Version = "5.2.3790.1830 (srv03_sp1_rtm.050324-1447)"
User Name	SIEBW2K3\sadmin
Time Zone	Pacific Standard Time
Total Physical Memory	24,574.91 MB
Available Physical Memory	2.67 GB
Total Virtual Memory	3.70 GB
Available Virtual Memory	2.82 GB
Page File Space	.00 KB

Dual-Core Intel Xeon MP CPUs

Item	Value
OS Name	Microsoft(R) Windows(R) Server 2003, Enterprise Edition
Version	5.2.3790 Service Pack 1 Build 3790
Other OS Description	Not Available
OS Manufacturer	Microsoft Corporation
System Name	EL9-92-154
System Manufacturer	IBM
System Model	eserver xSeries 366-[4444aaa]-
System Type	X86-based PC
Processor	X86 Family 15 Model 4 Stepping 8 GenuineIntel ~3003 Mhz
Processor	X86 Family 15 Model 4 Stepping 8 GenuineIntel ~3002 Mhz
Processor	X86 Family 15 Model 4 Stepping 8 GenuineIntel ~3002 Mhz
Processor	X86 Family 15 Model 4 Stepping 8 GenuineIntel ~3003 Mhz
Processor	X86 Family 15 Model 4 Stepping 8 GenuineIntel ~3002 Mhz
Processor	X86 Family 15 Model 4 Stepping 8 GenuineIntel ~3003 Mhz
Processor	X86 Family 15 Model 4 Stepping 8 GenuineIntel ~3002 Mhz
Processor	X86 Family 15 Model 4 Stepping 8 GenuineIntel ~3003 Mhz
Processor	X86 Family 15 Model 4 Stepping 8 GenuineIntel ~3003 Mhz
Processor	X86 Family 15 Model 4 Stepping 8 GenuineIntel ~3003 Mhz
Processor	X86 Family 15 Model 4 Stepping 8 GenuineIntel ~3003 Mhz
Processor	X86 Family 15 Model 4 Stepping 8 GenuineIntel ~3002 Mhz
Processor	X86 Family 15 Model 4 Stepping 8 GenuineIntel ~3002 Mhz
Processor	X86 Family 15 Model 4 Stepping 8 GenuineIntel ~3003 Mhz
Processor	X86 Family 15 Model 4 Stepping 8 GenuineIntel ~3003 Mhz
Processor	X86 Family 15 Model 4 Stepping 8 GenuineIntel ~3002 Mhz
BIOS Version/Date	IBM -[TESTBLDUS-1.05]-, 8/8/2005
SMBIOS Version	2.3
Windows Directory	C:\WINDOWS
System Directory	C:\WINDOWS\system32
Boot Device	\Device\HarddiskVolume1
Locale	United States
Hardware Abstraction Layer	Version = "5.2.3790.1830 (srv03_sp1_rtm.050324-1447)"
User Name	SIEBW2K3\sadmin
Time Zone	Pacific Standard Time
Total Physical Memory	24,574.90 MB
Available Physical Memory	2.53 GB
Total Virtual Memory	3.70 GB
Available Virtual Memory	2.70 GB
Page File Space	.00 KB

Appendix B: x460 System and Windows Server 2003 EE Configuration Information

Configuration data that appears here was captured using the Windows **msinfo32** utility.

Single-Core Intel Xeon MP CPUs (One Chassis)

Item	Value
OS Name	Microsoft(R) Windows(R) Server 2003, Enterprise Edition
Version	5.2.3790 Service Pack 1 Build 3790
Other OS Description	Not Available
OS Manufacturer	Microsoft Corporation
System Name	EL9-92-154
System Manufacturer	IBM
System Model	eserver xSeries 460-[88721RZ]-
System Type	X86-based PC
Processor	x86 Family 15 Model 4 Stepping 1 GenuineIntel ~3336 Mhz
Processor	x86 Family 15 Model 4 Stepping 1 GenuineIntel ~3336 Mhz
Processor	x86 Family 15 Model 4 Stepping 1 GenuineIntel ~3336 Mhz
Processor	x86 Family 15 Model 4 Stepping 1 GenuineIntel ~3336 Mhz
Processor	x86 Family 15 Model 4 Stepping 1 GenuineIntel ~3336 Mhz
Processor	x86 Family 15 Model 4 Stepping 1 GenuineIntel ~3336 Mhz
Processor	x86 Family 15 Model 4 Stepping 1 GenuineIntel ~3336 Mhz
Processor	x86 Family 15 Model 4 Stepping 1 GenuineIntel ~3336 Mhz
BIOS Version/Date	IBM -[ZUE139AUS-1.05]-, 8/2/2005
SMBIOS Version	2.3
Windows Directory	C:\WINDOWS
System Directory	C:\WINDOWS\system32
Boot Device	\Device\HarddiskVolume1
Locale	United States
Hardware Abstraction Layer	Version = "5.2.3790.1830 (srv03_sp1_rtm.050324-1447)"
User Name	SIEBW2K3\sadmin
Time Zone	Pacific Standard Time
Total Physical Memory	16,382.91 MB
Available Physical Memory	2.87 GB
Total Virtual Memory	3.76 GB
Available Virtual Memory	2.99 GB
Page File Space	.00 KB

Single-Core Intel Xeon MP CPUs (Two Chassis, Single Partition)

Item	Value
OS Name	Microsoft(R) Windows(R) Server 2003, Enterprise Edition
Version	5.2.3790 Service Pack 1 Build 3790
Other OS Description	Not Available
OS Manufacturer	Microsoft Corporation
System Name	EL9-92-154
System Manufacturer	IBM
System Model	eserver xSeries 460-[88721RZ]-
System Type	X86-based PC
Processor	x86 Family 15 Model 4 Stepping 1 GenuineIntel ~3336 Mhz
Processor	x86 Family 15 Model 4 Stepping 1 GenuineIntel ~3336 Mhz
Processor	x86 Family 15 Model 4 Stepping 1 GenuineIntel ~3336 Mhz
Processor	x86 Family 15 Model 4 Stepping 1 GenuineIntel ~3336 Mhz
Processor	x86 Family 15 Model 4 Stepping 1 GenuineIntel ~3336 Mhz
Processor	x86 Family 15 Model 4 Stepping 1 GenuineIntel ~3336 Mhz
Processor	x86 Family 15 Model 4 Stepping 1 GenuineIntel ~3335 Mhz
Processor	x86 Family 15 Model 4 Stepping 1 GenuineIntel ~3336 Mhz
Processor	x86 Family 15 Model 4 Stepping 1 GenuineIntel ~3336 Mhz
Processor	x86 Family 15 Model 4 Stepping 1 GenuineIntel ~3335 Mhz
Processor	x86 Family 15 Model 4 Stepping 1 GenuineIntel ~3336 Mhz
Processor	x86 Family 15 Model 4 Stepping 1 GenuineIntel ~3335 Mhz
Processor	x86 Family 15 Model 4 Stepping 1 GenuineIntel ~3336 Mhz
Processor	x86 Family 15 Model 4 Stepping 1 GenuineIntel ~3336 Mhz
Processor	x86 Family 15 Model 4 Stepping 1 GenuineIntel ~3336 Mhz
Processor	x86 Family 15 Model 4 Stepping 1 GenuineIntel ~3336 Mhz
BIOS Version/Date	IBM -[ZUE139AUS-1.05]-, 8/2/2005
SMBIOS Version	2.3
Windows Directory	C:\WINDOWS
System Directory	C:\WINDOWS\system32
Boot Device	\Device\HarddiskVolume1
Locale	United States
Hardware Abstraction Layer	Version = "5.2.3790.1830 (srv03_sp1_rtm.050324-1447)"
User Name	SIEBW2K3\sadmin
Time Zone	Pacific Standard Time
Total Physical Memory	32,254.87 MB
Available Physical Memory	1.86 GB
Total Virtual Memory	3.15 GB
Available Virtual Memory	2.02 GB
Page File Space	.00 KB

Appendix C: IBM Intel Performance Study Summary Results

In Table 1 through Table 3, the results have been color-coded to allow easier identification of each result set on the graphs in Graph 1 and Graph 2 that follow at the end of the appendix.

x366 System Single-Core Intel Xeon MP CPUs

The data in Table 1 was captured for the x366 system fitted with 4x single-core, 3.6 GHz Intel Xeon MP CPUs. This was a single-chassis configuration which had 4x 3.6 GHz CPU Cores.

Users	No. of CPUs	No. of CPU Cores	Total CPU%	User CPU%	System CPU%	Hits/Sec	Avg Response	Users/CPU Core (@100%)	Users/CPU Core (actual)	RAM/User (MB)
500	4	4	9.70	9.01	0.68	58.80	0.04	1289	125	7.07
700	4	4	13.76	12.72	1.04	82.90	0.05	1272	175	6.44
900	4	4	17.99	16.72	1.27	106.76	0.05	1251	225	6.12
1100	4	4	25.03	22.98	2.05	129.55	0.05	1099	275	5.93
1300	4	4	29.72	27.43	2.28	153.33	0.05	1094	325	5.80
1500	4	4	37.21	34.38	2.82	176.96	0.05	1008	375	5.68
1700	4	4	42.23	38.98	3.24	201.09	0.06	1006	425	5.59
1900	4	4	51.91	47.86	4.04	224.41	0.06	915	475	5.55
2100	4	4	58.11	53.67	4.44	249.15	0.06	904	525	5.41
2300	4	4	70.71	65.48	5.22	272.38	0.07	813	575	5.41
2500	4	4	78.83	72.90	5.92	295.48	0.07	793	625	5.40
2700	4	4	89.74	83.08	6.66	318.21	0.11	752	675	5.39
2800	4	4	95.10	87.89	7.21	328.60	0.18	736	700	5.45

Table 1: Single-Core Intel Xeon MP CPUs

x366 System Dual-Core Intel Xeon MP CPUs

The data in Table 2 was captured for the x366 system fitted with 4x dual-core, 3.0 GHz Intel Xeon MP CPUs. This was a single-chassis configuration which had 8x 3.0 GHz CPU Cores.

Users	No. of CPUs	No. of CPU Cores	Total CPU%	User CPU%	System CPU%	Hits/Sec	Avg Response	Users/CPU Core (@100%)	Users/CPU Core (actual)	RAM/User (MB)
500	4	8	5.56	5.09	0.47	58.80	0.04	1124	63	7.08
700	4	8	7.80	7.21	0.58	82.21	0.04	1122	88	6.48
900	4	8	10.64	9.76	0.87	105.86	0.04	1058	113	6.13
1100	4	8	12.91	11.89	1.01	130.36	0.05	1065	138	5.92
1300	4	8	15.78	14.53	1.25	153.40	0.05	1030	163	5.78
1500	4	8	18.84	17.34	1.49	177.00	0.05	995	188	5.70
1700	4	8	21.87	20.10	1.76	201.38	0.05	972	213	5.62
1900	4	8	24.26	22.23	2.03	224.57	0.05	979	238	5.55
2100	4	8	27.48	25.20	2.28	249.29	0.05	955	263	5.50
2300	4	8	30.27	27.66	2.61	272.41	0.05	950	288	5.46
2500	4	8	35.21	32.23	2.98	297.30	0.06	888	313	5.44
2700	4	8	38.29	34.86	3.43	319.78	0.06	881	338	5.42
2900	4	8	42.45	38.69	3.76	343.30	0.06	854	363	5.37
3100	4	8	46.74	42.23	4.51	367.04	0.06	829	388	5.37
3300	4	8	50.49	45.82	4.67	391.23	0.06	817	413	5.36
3500	4	8	56.37	50.76	5.60	415.87	0.07	776	438	5.34
3700	4	8	59.92	53.77	6.15	438.67	0.07	772	463	5.34
3900	4	8	67.28	60.51	6.77	461.90	0.07	725	488	5.33
4100	4	8	73.20	66.00	7.20	486.22	0.07	700	513	5.27
4300	4	8	80.64	72.33	8.30	509.55	0.08	667	538	5.27
4500	4	8	86.61	77.62	8.98	531.92	0.10	649	563	5.20

Table 2: Dual-Core Intel Xeon MP CPUs

x460 System Single-Core Intel Xeon MP CPUs (Two Chassis, Single Partition)

The data in Table 3 was captured for the x460 system fitted with 8x single-core, 3.3 GHz Intel Xeon MP CPUs. This was a two-chassis, single-partition configuration which had 8x 3.3 GHz CPU Cores (4x single-core CPUs per chassis).

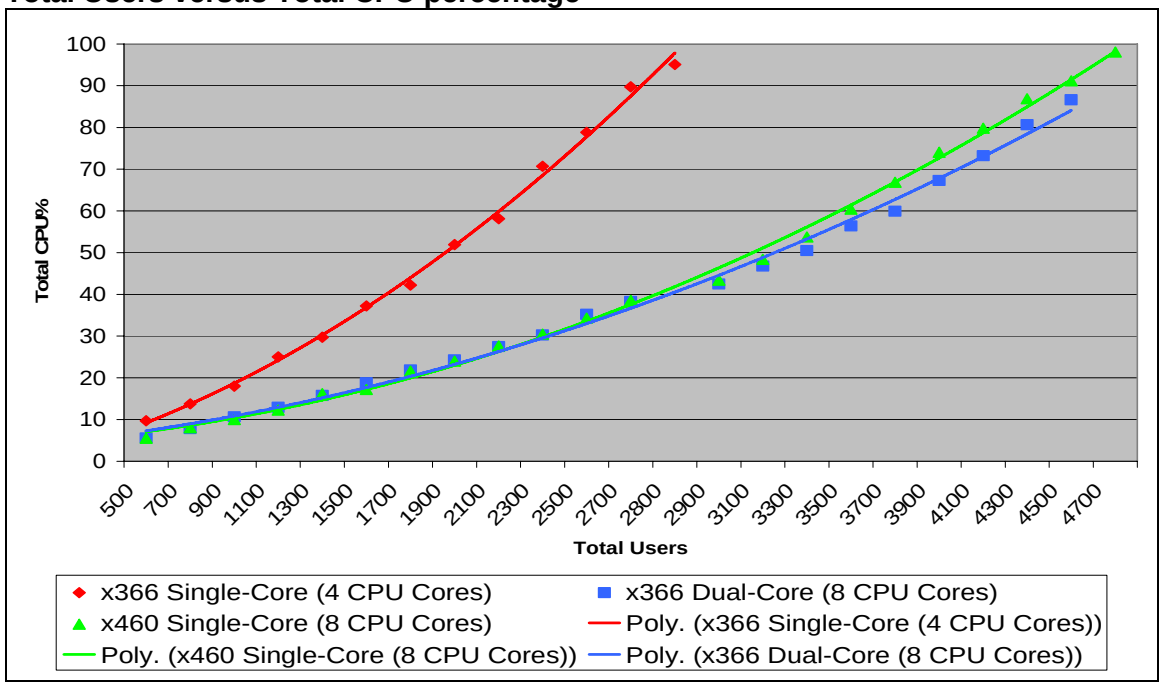
Users	No. of CPUs	No. of CPU Cores	Total CPU%	User CPU%	System CPU%	Hits/Sec	Avg Response	Users/CPU Core (@100%)	Users/CPU Core (actual)	RAM/User (MB)
500	8	8	5.54	4.89	0.65	58.71	0.04	1129	63	6.97
700	8	8	8.06	7.16	0.90	82.66	0.04	1085	88	6.40
900	8	8	9.97	8.84	1.14	106.44	0.04	1128	113	6.09
1100	8	8	12.26	10.87	1.38	130.60	0.04	1122	138	5.87
1300	8	8	16.19	14.45	1.74	153.90	0.04	1003	163	5.74
1500	8	8	17.22	15.15	2.07	177.49	0.04	1089	188	5.67
1700	8	8	21.58	19.18	2.40	200.77	0.04	985	213	5.59
1900	8	8	23.98	21.22	2.75	224.36	0.05	991	238	5.51
2100	8	8	27.63	24.34	3.28	248.78	0.05	950	263	5.47
2300	8	8	30.47	26.72	3.75	272.91	0.05	943	288	5.42
2500	8	8	34.28	30.09	4.18	296.02	0.05	912	313	5.40
2700	8	8	38.56	33.89	4.67	320.93	0.05	875	338	5.38
2900	8	8	43.40	37.92	5.48	343.96	0.05	835	363	5.37
3100	8	8	48.37	42.28	6.09	367.29	0.06	801	388	5.34
3300	8	8	53.73	46.67	7.05	391.96	0.06	768	413	5.35
3500	8	8	60.42	52.59	7.83	415.07	0.06	724	438	5.33
3700	8	8	66.84	58.38	8.46	439.39	0.07	692	463	5.31
3900	8	8	74.03	64.09	9.93	462.78	0.07	659	488	5.30
4100	8	8	79.82	69.45	10.37	486.06	0.08	642	513	5.29
4300	8	8	86.88	75.50	11.38	509.19	0.09	619	538	5.29
4500	8	8	91.15	78.89	12.25	532.36	0.10	617	563	5.27
4700	8	8	98.06	84.76	13.30	552.84	0.19	599	588	5.26

Table 3: Single-Core Intel Xeon MP CPUs (Two Chassis, Single Partition)

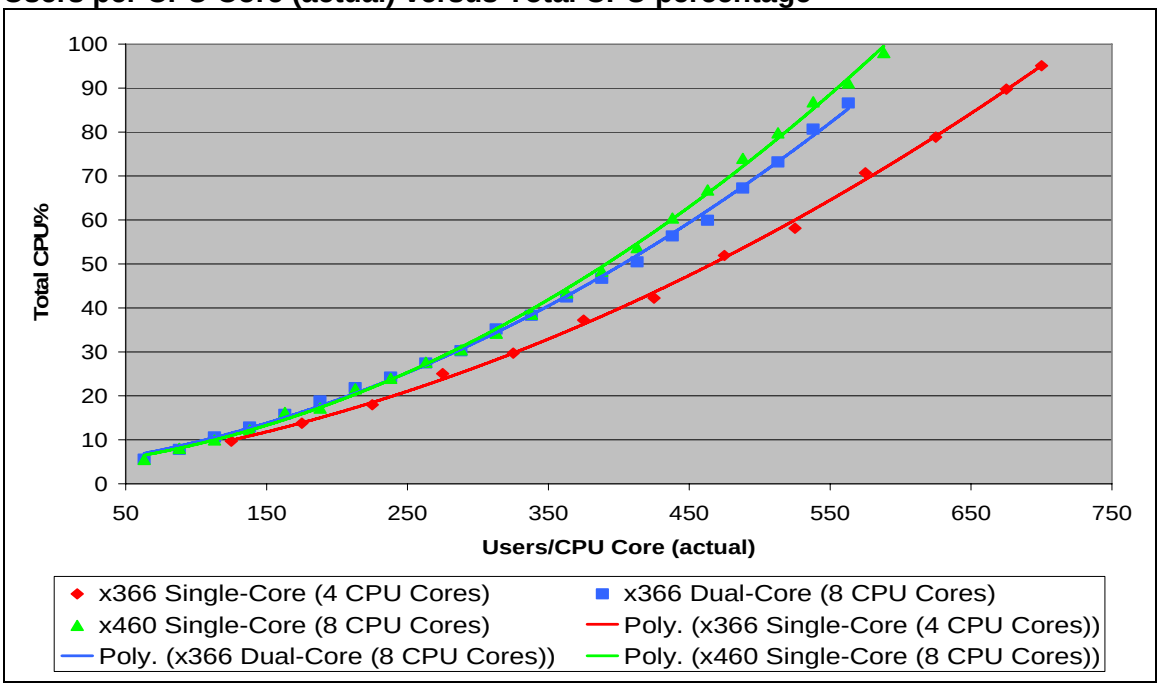
Graphical Representations of Summary Results

The graphs in Graph 1 and Graph 2 show the summary results.

Total Users versus Total CPU percentage



Graph 1: Total Users versus Total CPU Percentage



Graph 2: Users per CPU Core (actual) versus Total CPU percentage

Appendix D: IBM Intel Performance Study Siebel Application Server Configuration Examples

Before each data point was captured, the Siebel 7.7 Application Server was reconfigured to support slightly more Siebel users than targeted by the LoadRunner scenario. As a rule of thumb, the application server was always configured for 100 to 200 more users than the scenario called for.

Many test runs were performed to determine the best Siebel component base configuration to use. Once determined, all future configurations were scaled up versions of the base, using the same option settings and parameter ratios.

The following sections illustrate the Siebel Application Server configurations that were applied to achieve the best run for each xSeries hardware category.

x366 System Single-Core Intel Xeon MP CPUs

Siebel Application Server Configuration

The following is the Korn shell script used to configure the Siebel Application Server in preparation for a 2,800 user run:

```
#!/bin/ksh
# Config Script Version Number: 1

function apply_config {
#
# Global Configuration Variables Used To Determine Parameters.
# Target No. Users: 3000
# No. Siebel App. Servers: 1
# No. Web Servers: 1
# App. Server CPU Bound?: False
# AOM SQL Cursor Cache: 0
# AOM SQL Data Cache: 0
# App. Server OSDF Latches Factor: 3300
# Web Client Threads: 3000
#

#
# Siebel Service Request Broker
#
COMPONENT=SRBroker
echo "change param MaxTasks=100 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MinMTServers=1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MaxMTServers=1 for comp "$COMPONENT >> $COMMAND_FILE

#
# Siebel Server Data Source
#
COMPONENT=ServerDataSrc
echo "change param DSMaxCachedCursors=0 for named subsystem "$COMPONENT >> $COMMAND_FILE
echo "change param DSMaxCachedDataSets=0 for named subsystem "$COMPONENT >> $COMMAND_FILE

#
# AOM Specific Configuration Variables Used To Determine Parameters.
# AOM1 Component Name: FINSObjMgr_enu
# No. Tasks per Thread: 1
# Thread Pooling Enabled?: False
# Thread Affinity Enabled?: True
# No. Users per DB Connection: 1
# Desired Users per AOM: 100
#
COMPONENT=FINSObjMgr_enu
echo "change param MaxTasks=3000 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MinMTServers=30 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MaxMTServers=30 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MinSharedDbConns=-1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MaxSharedDbConns=-1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MinTrxDbConns=-1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MinPoolThreads=0 for comp "$COMPONENT >> $COMMAND_FILE
```

```
echo "change param MaxPoolThreads=0 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param SessPerSisnConn=20 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param CommEnable=False for comp "$COMPONENT >> $COMMAND_FILE
echo "change param CommConfigManager=True for comp "$COMPONENT >> $COMMAND_FILE
echo "change param UseThreadPool=False for comp "$COMPONENT >> $COMMAND_FILE
echo "change param ThreadAffinity=True for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MemProtection=True for comp "$COMPONENT >> $COMMAND_FILE
echo "change EvtLogLvl EventContext=1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change EvtLogLvl ObjMgrSqlLog=1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change EvtLogLvl SQLFlags=0 for service "$COMPONENT >> $COMMAND_FILE

#
# AOM Specific Configuration Variables Used To Determine Parameters.
# AOM2 Component Name: eChannelObjMgr_enu
# No. Tasks per Thread: 1
# Thread Pooling Enabled?: False
# Thread Affinity Enabled?: True
# No. Users per DB Connection: 1
# Desired Users per AOM: 100
#
COMPONENT=eChannelObjMgr_enu
echo "change param MaxTasks=0 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MinMTServers=1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MaxMTServers=1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MinSharedDbConns=-1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MaxSharedDbConns=-1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MinTrxDbConns=-1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MinPoolThreads=0 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MaxPoolThreads=0 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param SessPerSisnConn=20 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param CommEnable=False for comp "$COMPONENT >> $COMMAND_FILE
echo "change param CommConfigManager=True for comp "$COMPONENT >> $COMMAND_FILE
echo "change param UseThreadPool=False for comp "$COMPONENT >> $COMMAND_FILE
echo "change param ThreadAffinity=True for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MemProtection=True for comp "$COMPONENT >> $COMMAND_FILE
echo "change EvtLogLvl EventContext=1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change EvtLogLvl ObjMgrSqlLog=1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change EvtLogLvl SQLFlags=0 for service "$COMPONENT >> $COMMAND_FILE

#
# AOM Specific Configuration Variables Used To Determine Parameters.
# AOM3 Component Name: WfProcMgr
# No. Tasks per Thread: 1
# Thread Pooling Enabled?: False
# Thread Affinity Enabled?: True
# No. Users per DB Connection: 2
# Desired Users per AOM: 100
#
COMPONENT=WfProcMgr
echo "change param MaxTasks=0 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MinMTServers=1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MaxMTServers=1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MinSharedDbConns=0 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MaxSharedDbConns=0 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MinTrxDbConns=0 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MinPoolThreads=0 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MaxPoolThreads=0 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param SessPerSisnConn=0 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param HonorMaxTasks=True for comp "$COMPONENT >> $COMMAND_FILE
echo "change param UsePrivatePort=True for comp "$COMPONENT >> $COMMAND_FILE
echo "change param BypassHandler=True for comp "$COMPONENT >> $COMMAND_FILE
echo "change param ModelCacheMax=84 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MemProtection=True for comp "$COMPONENT >> $COMMAND_FILE
echo "change EvtLogLvl EventContext=1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change EvtLogLvl ObjMgrSqlLog=1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change EvtLogLvl SQLFlags=0 for service "$COMPONENT >> $COMMAND_FILE

#
# Apply the configuration changes to the Siebel Application Server
#
"c:/sea77/siebsrvr/bin/srvrmgr" $CONNECT_STRING <<EOF
read $COMMAND_FILE
quit
EOF

printf "\n'srvrmgr' Connect String Used: %s\n" "$CONNECT_STRING"
printf "Application Server Configured: %s\n\n" "$COMPUTERNAME"
}

################################################################################
```

```
################################################################################
#
# Main Script Starts Here
#
################################################################################
################################################################################

CONNECT_STRING="-e siebel -g $COMPUTERNAME -u sadmin -p sadmin"
COMMAND_FILE=c:/temp/srvrmgr.$$
OUTPUT_FILE=c:/temp/srvrmgr.out.$$

if [[ -f $COMMAND_FILE ]]; then
   rm $COMMAND_FILE
fi

if [[ -f $OUTPUT_FILE ]]; then
   rm $OUTPUT_FILE
fi

#
# Specify the App Server we want to configure
#
echo "set server "$COMPUTERNAME >> $COMMAND_FILE

#
# Execute the 'set server' command in 'srvrmgr' and capture the OUTPUT
#
"c:/sea77/siebsrvr/bin/srvrmgr" $CONNECT_STRING <<EOF >$OUTPUT_FILE
read $COMMAND_FILE
quit
EOF

#
# GREP the OUTPUT from the 'srvrmgr' command for our App Server Name in the
# PROMPT. The PROMPT will take the format 'srvrmgr:<App Server Name>', if
# the 'set server' command was successful
#
if [[ -z "`grep \"srvrmgr:$COMPUTERNAME\" $OUTPUT_FILE`" ]]; then
   printf "\n\nERROR! - Siebel App Server '%s' Not Found In Enterprise '%s'!\n\n"
"$COMPUTERNAME"  "siebel"
else
   #
   # App Server exists! Apply the config to it. The FUNCTION continues to APPEND
   # the config details to the COMMAND_FILE, which at the moment, contains just
   # the valid 'set server' command
   #
   apply_config
fi

rm $COMMAND_FILE
rm $OUTPUT_FILE
```

Siebel Run-time Processes

The following is a custom display of the Siebel Application Server processes that were running before the start of the 2,800 user run:

```
COMMAND    PID      SC     THCNT  %CPU  %MEM  VSZ      TASK_ID  SIEBEL_COMPONENT
siebmtshmw 10552    0      6      0.0   0.0   45892    0        FINSObjMgr_enu
siebmtshmw 11568    0      6      0.0   0.0   45892    0        FINSObjMgr_enu
siebmtshmw 11696    0      6      0.0   0.0   45896    0        FINSObjMgr_enu
siebmtshmw 12028    0      6      0.0   0.0   45892    0        FINSObjMgr_enu
siebmtshmw 12080    0      6      0.0   0.0   45892    0        FINSObjMgr_enu
siebmtshmw 12888    0      6      0.0   0.0   45892    0        FINSObjMgr_enu
siebmtshmw 13812    0      6      0.0   0.0   45892    0        FINSObjMgr_enu
siebmtshmw 14116    0      6      0.0   0.0   45892    0        FINSObjMgr_enu
siebmtshmw 15176    0      6      0.0   0.0   45892    0        FINSObjMgr_enu
siebmtshmw 2592     0      6      0.0   0.0   45896    0        FINSObjMgr_enu
siebmtshmw 2816     0      6      0.0   0.0   45892    0        FINSObjMgr_enu
siebmtshmw 2868     0      6      0.0   0.0   45892    0        FINSObjMgr_enu
siebmtshmw 3524     0      6      0.0   0.0   45884    0        FINSObjMgr_enu
siebmtshmw 3744     0      6      0.0   0.0   45888    0        FINSObjMgr_enu
siebmtshmw 4108     0      6      0.0   0.0   45892    0        FINSObjMgr_enu
siebmtshmw 4372     0      6      0.0   0.0   45896    0        FINSObjMgr_enu
siebmtshmw 4396     0      6      0.0   0.0   45896    0        FINSObjMgr_enu
siebmtshmw 4592     0      6      0.0   0.0   45892    0        FINSObjMgr_enu
siebmtshmw 4888     0      7      0.0   0.0   45880    0        FINSObjMgr_enu
siebmtshmw 5220     0      6      0.0   0.0   45892    0        FINSObjMgr_enu
siebmtshmw 5460     0      6      0.0   0.0   45888    0        FINSObjMgr_enu
siebmtshmw 6408     0      6      0.0   0.0   45896    0        FINSObjMgr_enu
siebmtshmw 6972     0      6      0.0   0.0   45892    0        FINSObjMgr_enu
siebmtshmw 7332     0      6      0.0   0.0   45884    0        FINSObjMgr_enu
siebmtshmw 7416     0      7      0.0   0.0   45880    0        FINSObjMgr_enu
siebmtshmw 8488     0      6      0.0   0.0   45888    0        FINSObjMgr_enu
siebmtshmw 9420     0      7      0.0   0.0   45884    0        FINSObjMgr_enu
siebmtshmw 9484     0      7      0.0   0.0   45876    0        FINSObjMgr_enu
siebmtshmw 9692     0      6      0.0   0.0   45888    0        FINSObjMgr_enu
siebmtshmw 9812     0      6      0.0   0.0   45896    0        FINSObjMgr_enu
siebmtsh   2468     0      25     0.0   0.0   11056    0        FSMSrvr
siebproc   13720    0      1      0.0   0.0   8068     0        SCBroker
siebmtsh   3568     0      16     0.0   0.0   20184    0        SRBroker
siebmtsh   652      0      8      0.0   0.0   96516    0        SRProc
siebmtshmw 3836     0      6      0.0   0.0   142152   0        WfProcMgr
siebmtshmw 12716    0      6      0.0   0.0   20012    0        eChannelObjMgr_enu

SIEBEL_COMPONENT               CNT    THCNT  %CPU    TOT_MEM   AVG_MEM
================               ====   =====  ====    =======  ========
FINSObjMgr_enu                 30     184     0.0    1376704   45890.1
FSMSrvr                        1      25      0.0      11056   11056.0
SCBroker                       1      1       0.0       8068    8068.0
SRBroker                       1      16      0.0      20184   20184.0
SRProc                         1      8       0.0      96516   96516.0
WfProcMgr                      1      6       0.0     142152  142152.0
eChannelObjMgr_enu             1      6       0.0      20012   20012.0
                               ====   =====  ====     ======  ========
           Cumulative          36     246       0    1674692
```

The following are the Siebel Application Server processes that were running after the 2,800 user run:

```
COMMAND    PID      SC     THCNT  %CPU  %MEM  VSZ      TASK_ID  SIEBEL_COMPONENT
siebmtshmw 10552    0      7      0.0   0.0   408416   0        FINSObjMgr_enu
siebmtshmw 11568    0      7      0.0   0.0   406224   0        FINSObjMgr_enu
siebmtshmw 11696    0      7      0.0   0.0   409540   0        FINSObjMgr_enu
siebmtshmw 12028    0      7      0.0   0.0   404280   0        FINSObjMgr_enu
siebmtshmw 12080    0      7      0.0   0.0   408764   0        FINSObjMgr_enu
siebmtshmw 12888    0      7      0.0   0.0   409004   0        FINSObjMgr_enu
siebmtshmw 13812    0      7      0.0   0.0   405876   0        FINSObjMgr_enu
siebmtshmw 14116    0      7      0.0   0.0   405328   0        FINSObjMgr_enu
siebmtshmw 15176    0      7      0.0   0.0   407664   0        FINSObjMgr_enu
siebmtshmw 2592     0      7      0.0   0.0   405952   0        FINSObjMgr_enu
siebmtshmw 2816     0      7      0.0   0.0   405620   0        FINSObjMgr_enu
siebmtshmw 2868     0      7      0.0   0.0   405624   0        FINSObjMgr_enu
siebmtshmw 3524     0      7      0.0   0.0   410236   0        FINSObjMgr_enu
siebmtshmw 3744     0      7      0.0   0.0   405064   0        FINSObjMgr_enu
siebmtshmw 4108     0      7      0.0   0.0   409420   0        FINSObjMgr_enu
siebmtshmw 4372     0      7      0.0   0.0   409316   0        FINSObjMgr_enu
siebmtshmw 4396     0      7      0.0   0.0   407588   0        FINSObjMgr_enu
siebmtshmw 4592     0      7      0.0   0.0   406832   0        FINSObjMgr_enu
siebmtshmw 4888     0      7      0.0   0.0   406816   0        FINSObjMgr_enu
siebmtshmw 5220     0      7      0.0   0.0   409108   0        FINSObjMgr_enu
siebmtshmw 5460     0      7      0.0   0.0   408764   0        FINSObjMgr_enu
siebmtshmw 6408     0      7      0.0   0.0   406740   0        FINSObjMgr_enu
siebmtshmw 6972     0      7      0.0   0.0   406012   0        FINSObjMgr_enu
siebmtshmw 7332     0      7      0.0   0.0   407696   0        FINSObjMgr_enu
siebmtshmw 7416     0      7      0.0   0.0   405844   0        FINSObjMgr_enu
siebmtshmw 8488     0      7      0.0   0.0   408176   0        FINSObjMgr_enu
siebmtshmw 9420     0      7      0.0   0.0   406968   0        FINSObjMgr_enu
siebmtshmw 9484     0      7      0.0   0.0   407684   0        FINSObjMgr_enu
siebmtshmw 9692     0      7      0.0   0.0   411352   0        FINSObjMgr_enu
siebmtshmw 9812     0      7      0.0   0.0   407512   0        FINSObjMgr_enu
siebmtsh   2468     0      25     0.0   0.0   11056    0        FSMSrvr
siebproc   13720    0      1      0.0   0.0   8416     0        SCBroker
siebmtsh   3568     0      42     0.0   0.0   24180    0        SRBroker
siebmtsh   652      0      8      0.0   0.0   96520    0        SRProc
siebmtshmw 3836     0      6      0.0   0.0   142152   0        WfProcMgr
siebmtshmw 12716    0      6      0.0   0.0   20012    0        eChannelObjMgr_enu

SIEBEL_COMPONENT              CNT   THCNT  %CPU    TOT_MEM   AVG_MEM
================              ====  =====  ====    =======   ========
FINSObjMgr_enu                30    210     0.0   12223420  407447.3
FSMSrvr                       1     25      0.0      11056   11056.0
SCBroker                      1     1       0.0       8416    8416.0
SRBroker                      1     42      0.0      24180   24180.0
SRProc                        1     8       0.0      96520   96520.0
WfProcMgr                     1     6       0.0     142152  142152.0
eChannelObjMgr_enu            1     6       0.0      20012   20012.0
                              ====  =====  ====     ======   ========
           Cumulative         36    298       0   12525756
```

x366 System Dual-Core Intel Xeon MP CPUs

Siebel Application Server Configuration

The following is the Korn shell script used to configure the Siebel Application Server in preparation for a 4,500 user run:

```
#!/bin/ksh
# Config Script Version Number: 1

function apply_config {
#
# Global Configuration Variables Used To Determine Parameters.
# Target No. Users: 4600
# No. Siebel App. Servers: 1
# No. Web Servers: 1
# App. Server CPU Bound?: False
# AOM SQL Cursor Cache: 0
# AOM SQL Data Cache: 0
# App. Server OSDF Latches Factor: 5060
# Web Client Threads: 4600
#

#
# Siebel Service Request Broker
#
COMPONENT=SRBroker
echo "change param MaxTasks=100 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MinMTServers=1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MaxMTServers=1 for comp "$COMPONENT >> $COMMAND_FILE

#
# Siebel Server Data Source
#
COMPONENT=ServerDataSrc
echo "change param DSMaxCachedCursors=0 for named subsystem "$COMPONENT >> $COMMAND_FILE
echo "change param DSMaxCachedDataSets=0 for named subsystem "$COMPONENT >> $COMMAND_FILE

#
# AOM Specific Configuration Variables Used To Determine Parameters.
# AOM1 Component Name: FINSObjMgr_enu
# No. Tasks per Thread: 1
# Thread Pooling Enabled?: False
# Thread Affinity Enabled?: True
# No. Users per DB Connection: 1
# Desired Users per AOM: 100
#
COMPONENT=FINSObjMgr_enu
echo "change param MaxTasks=4600 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MinMTServers=46 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MaxMTServers=46 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MinSharedDbConns=-1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MaxSharedDbConns=-1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MinTrxDbConns=-1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MinPoolThreads=0 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MaxPoolThreads=0 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param SessPerSisnConn=20 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param CommEnable=False for comp "$COMPONENT >> $COMMAND_FILE
echo "change param CommConfigManager=True for comp "$COMPONENT >> $COMMAND_FILE
echo "change param UseThreadPool=False for comp "$COMPONENT >> $COMMAND_FILE
echo "change param ThreadAffinity=True for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MemProtection=True for comp "$COMPONENT >> $COMMAND_FILE
echo "change EvtLogLvl EventContext=1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change EvtLogLvl ObjMgrSqlLog=1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change EvtLogLvl SQLFlags=0 for service "$COMPONENT >> $COMMAND_FILE

#
# AOM Specific Configuration Variables Used To Determine Parameters.
# AOM2 Component Name: eChannelObjMgr_enu
# No. Tasks per Thread: 1
# Thread Pooling Enabled?: False
# Thread Affinity Enabled?: True
# No. Users per DB Connection: 1
# Desired Users per AOM: 100
#
COMPONENT=eChannelObjMgr_enu
echo "change param MaxTasks=0 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MinMTServers=1 for comp "$COMPONENT >> $COMMAND_FILE
```

```
echo "change param MaxMTServers=1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MinSharedDbConns=-1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MaxSharedDbConns=-1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MinTrxDbConns=-1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MinPoolThreads=0 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MaxPoolThreads=0 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param SessPerSisnConn=20 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param CommEnable=False for comp "$COMPONENT >> $COMMAND_FILE
echo "change param CommConfigManager=True for comp "$COMPONENT >> $COMMAND_FILE
echo "change param UseThreadPool=False for comp "$COMPONENT >> $COMMAND_FILE
echo "change param ThreadAffinity=True for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MemProtection=True for comp "$COMPONENT >> $COMMAND_FILE
echo "change EvtLogLvl EventContext=1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change EvtLogLvl ObjMgrSqlLog=1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change EvtLogLvl SQLFlags=0 for service "$COMPONENT >> $COMMAND_FILE

#
# AOM Specific Configuration Variables Used To Determine Parameters.
# AOM3 Component Name: WfProcMgr
# No. Tasks per Thread: 1
# Thread Pooling Enabled?: False
# Thread Affinity Enabled?: True
# No. Users per DB Connection: 2
# Desired Users per AOM: 100
#
COMPONENT=WfProcMgr
echo "change param MaxTasks=0 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MinMTServers=1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MaxMTServers=1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MinSharedDbConns=0 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MaxSharedDbConns=0 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MinTrxDbConns=0 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MinPoolThreads=0 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MaxPoolThreads=0 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param SessPerSisnConn=0 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param HonorMaxTasks=True for comp "$COMPONENT >> $COMMAND_FILE
echo "change param UsePrivatePort=True for comp "$COMPONENT >> $COMMAND_FILE
echo "change param BypassHandler=True for comp "$COMPONENT >> $COMMAND_FILE
echo "change param ModelCacheMax=84 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MemProtection=True for comp "$COMPONENT >> $COMMAND_FILE
echo "change EvtLogLvl EventContext=1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change EvtLogLvl ObjMgrSqlLog=1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change EvtLogLvl SQLFlags=0 for service "$COMPONENT >> $COMMAND_FILE

#
# Apply the configuration changes to the Siebel Application Server
#
"c:/sea77/siebsrvr/bin/srvrmgr" $CONNECT_STRING <<EOF
read $COMMAND_FILE
quit
EOF

printf "\n'srvrmgr' Connect String Used: %s\n" "$CONNECT_STRING"
printf "Application Server Configured: %s\n\n" "$COMPUTERNAME"
}

################################################################################
################################################################################
#
# Main Script Starts Here
#
################################################################################
################################################################################

CONNECT_STRING="-e siebel -g $COMPUTERNAME -u sadmin -p sadmin"
COMMAND_FILE=c:/temp/srvrmgr.$$
OUTPUT_FILE=c:/temp/srvrmgr.out.$$

if [[ -f $COMMAND_FILE ]]; then
   rm $COMMAND_FILE
fi

if [[ -f $OUTPUT_FILE ]]; then
   rm $OUTPUT_FILE
fi

#
# Specify the App Server we want to configure
#
echo "set server "$COMPUTERNAME >> $COMMAND_FILE
```

```
#
# Execute the 'set server' command in 'srvrmgr' and capture the OUTPUT
#
"c:/sea77/siebsrvr/bin/srvrmgr" $CONNECT_STRING <<EOF >$OUTPUT_FILE
read $COMMAND_FILE
quit
EOF

#
# GREP the OUTPUT from the 'srvrmgr' command for our App Server Name in the
# PROMPT. The PROMPT will take the format 'srvrmgr:<App Server Name>', if
# the 'set server' command was successful
#
if [[ -z "`grep \"srvrmgr:$COMPUTERNAME\" $OUTPUT_FILE`" ]]; then
   printf "\n\nERROR! - Siebel App Server '%s' Not Found In Enterprise '%s'!\n\n"
"$COMPUTERNAME"  "siebel"
else
   #
   # App Server exists! Apply the config to it. The FUNCTION continues to APPEND
   # the config details to the COMMAND_FILE, which at the moment, contains just
   # the valid 'set server' command
   #
   apply_config
fi

rm $COMMAND_FILE
rm $OUTPUT_FILE
```

Siebel Run-time Processes

The following is a custom display of the Siebel Application Server processes that were running before the start of the 4,500 user run:

```
COMMAND    PID      SC     THCNT  %CPU  %MEM  VSZ      TASK_ID  SIEBEL_COMPONENT
siebmtshmw 1052     0      6      0.0   0.0   60056    0        FINSObjMgr_enu
siebmtshmw 2076     0      6      0.0   0.0   60052    0        FINSObjMgr_enu
siebmtshmw 2116     0      6      0.0   0.0   60052    0        FINSObjMgr_enu
siebmtshmw 2344     0      7      0.0   0.0   60076    0        FINSObjMgr_enu
siebmtshmw 2364     0      6      0.0   0.0   60052    0        FINSObjMgr_enu
siebmtshmw 2372     0      6      0.0   0.0   60052    0        FINSObjMgr_enu
siebmtshmw 2600     0      6      0.0   0.0   60052    0        FINSObjMgr_enu
siebmtshmw 2604     0      6      0.0   0.0   60052    0        FINSObjMgr_enu
siebmtshmw 2624     0      6      0.0   0.0   60060    0        FINSObjMgr_enu
siebmtshmw 272      0      6      0.0   0.0   60056    0        FINSObjMgr_enu
siebmtshmw 2760     0      6      0.0   0.0   60056    0        FINSObjMgr_enu
siebmtshmw 2836     0      6      0.0   0.0   60056    0        FINSObjMgr_enu
siebmtshmw 2864     0      6      0.0   0.0   60060    0        FINSObjMgr_enu
siebmtshmw 2904     0      7      0.0   0.0   60076    0        FINSObjMgr_enu
siebmtshmw 2936     0      6      0.0   0.0   60056    0        FINSObjMgr_enu
siebmtshmw 2996     0      6      0.0   0.0   60056    0        FINSObjMgr_enu
siebmtshmw 3076     0      6      0.0   0.0   60060    0        FINSObjMgr_enu
siebmtshmw 3132     0      6      0.0   0.0   60052    0        FINSObjMgr_enu
siebmtshmw 3176     0      6      0.0   0.0   60056    0        FINSObjMgr_enu
siebmtshmw 3248     0      6      0.0   0.0   60060    0        FINSObjMgr_enu
siebmtshmw 3272     0      6      0.0   0.0   60056    0        FINSObjMgr_enu
siebmtshmw 3352     0      6      0.0   0.0   60056    0        FINSObjMgr_enu
siebmtshmw 3356     0      6      0.0   0.0   60056    0        FINSObjMgr_enu
siebmtshmw 3404     0      6      0.0   0.0   60056    0        FINSObjMgr_enu
siebmtshmw 3416     0      6      0.0   0.0   60052    0        FINSObjMgr_enu
siebmtshmw 3440     0      6      0.0   0.0   60068    0        FINSObjMgr_enu
siebmtshmw 3500     0      6      0.0   0.0   60064    0        FINSObjMgr_enu
siebmtshmw 3528     0      6      0.0   0.0   60060    0        FINSObjMgr_enu
siebmtshmw 3552     0      6      0.0   0.0   60068    0        FINSObjMgr_enu
siebmtshmw 3636     0      6      0.0   0.0   60056    0        FINSObjMgr_enu
siebmtshmw 3640     0      6      0.0   0.0   60052    0        FINSObjMgr_enu
siebmtshmw 368      0      6      0.0   0.0   60056    0        FINSObjMgr_enu
siebmtshmw 3720     0      6      0.0   0.0   60060    0        FINSObjMgr_enu
siebmtshmw 3776     0      6      0.0   0.0   60064    0        FINSObjMgr_enu
siebmtshmw 380      0      6      0.0   0.0   60048    0        FINSObjMgr_enu
siebmtshmw 3864     0      6      0.0   0.0   60060    0        FINSObjMgr_enu
siebmtshmw 3928     0      6      0.0   0.0   60056    0        FINSObjMgr_enu
siebmtshmw 3984     0      7      0.0   0.0   60072    0        FINSObjMgr_enu
siebmtshmw 4028     0      6      0.0   0.0   60060    0        FINSObjMgr_enu
siebmtshmw 4060     0      7      0.0   0.0   60080    0        FINSObjMgr_enu
siebmtshmw 4092     0      6      0.0   0.0   60056    0        FINSObjMgr_enu
siebmtshmw 448      0      6      0.0   0.0   60048    0        FINSObjMgr_enu
siebmtshmw 640      0      7      0.0   0.0   60076    0        FINSObjMgr_enu
siebmtshmw 660      0      6      0.0   0.0   60060    0        FINSObjMgr_enu
siebmtshmw 692      0      6      0.0   0.0   60056    0        FINSObjMgr_enu
siebmtshmw 760      0      6      0.0   0.0   60052    0        FINSObjMgr_enu
siebmtsh   4068     0      25     0.0   0.0   11408    0        FSMSrvr
siebproc   764      0      1      0.0   0.0   8392     0        SCBroker
siebmtsh   2972     0      16     0.0   0.0   20648    0        SRBroker
siebmtsh   4076     0      8      0.0   0.0   97036    0        SRProc

SIEBEL_COMPONENT              CNT   THCNT  %CPU    TOT_MEM   AVG_MEM
================              ====  =====  ====    =======  ========
FINSObjMgr_enu                46    281     0.0    2762696   60058.6
FSMSrvr                       1     25      0.0      11408   11408.0
SCBroker                      1     1       0.0       8392    8392.0
SRBroker                      1     16      0.0      20648   20648.0
SRProc                        1     8       0.0      97036   97036.0
                              ====  =====  ====     ======  ========
           Cumulative         50    331       0    2900180
```

The following are the Siebel Application Server processes that were running after the 4,500 user run:

```
COMMAND    PID      SC      THCNT  %CPU  %MEM  VSZ      TASK_ID  SIEBEL_COMPONENT
siebmtshmw 1052     0       7      0.0   0.0   436128   0        FINSObjMgr_enu
siebmtshmw 2076     0       7      0.0   0.0   435428   0        FINSObjMgr_enu
siebmtshmw 2116     0       7      0.0   0.0   436276   0        FINSObjMgr_enu
siebmtshmw 2344     0       7      0.0   0.0   433448   0        FINSObjMgr_enu
siebmtshmw 2364     0       7      0.0   0.0   433284   0        FINSObjMgr_enu
siebmtshmw 2372     0       7      0.0   0.0   434604   0        FINSObjMgr_enu
siebmtshmw 2600     0       7      0.0   0.0   436720   0        FINSObjMgr_enu
siebmtshmw 2604     0       7      0.0   0.0   435612   0        FINSObjMgr_enu
siebmtshmw 2624     0       7      0.0   0.0   434408   0        FINSObjMgr_enu
siebmtshmw 272      0       7      0.0   0.0   433048   0        FINSObjMgr_enu
siebmtshmw 2760     0       7      0.0   0.0   434460   0        FINSObjMgr_enu
siebmtshmw 2836     0       7      0.0   0.0   433284   0        FINSObjMgr_enu
siebmtshmw 2864     0       7      0.0   0.0   435132   0        FINSObjMgr_enu
siebmtshmw 2904     0       7      0.0   0.0   436116   0        FINSObjMgr_enu
siebmtshmw 2936     0       7      0.0   0.0   435856   0        FINSObjMgr_enu
siebmtshmw 2996     0       7      0.0   0.0   434020   0        FINSObjMgr_enu
siebmtshmw 3076     0       7      0.0   0.0   435424   0        FINSObjMgr_enu
siebmtshmw 3132     0       7      0.0   0.0   434244   0        FINSObjMgr_enu
siebmtshmw 3176     0       7      0.0   0.0   432188   0        FINSObjMgr_enu
siebmtshmw 3248     0       7      0.0   0.0   434708   0        FINSObjMgr_enu
siebmtshmw 3272     0       7      0.0   0.0   434288   0        FINSObjMgr_enu
siebmtshmw 3352     0       7      0.0   0.0   435120   0        FINSObjMgr_enu
siebmtshmw 3356     0       7      0.0   0.0   434584   0        FINSObjMgr_enu
siebmtshmw 3404     0       7      0.0   0.0   436112   0        FINSObjMgr_enu
siebmtshmw 3416     0       7      0.0   0.0   433976   0        FINSObjMgr_enu
siebmtshmw 3440     0       7      0.0   0.0   437984   0        FINSObjMgr_enu
siebmtshmw 3500     0       7      0.0   0.0   432448   0        FINSObjMgr_enu
siebmtshmw 3528     0       7      0.0   0.0   436452   0        FINSObjMgr_enu
siebmtshmw 3552     0       7      0.0   0.0   434484   0        FINSObjMgr_enu
siebmtshmw 3636     0       7      0.0   0.0   432540   0        FINSObjMgr_enu
siebmtshmw 3640     0       7      0.0   0.0   433292   0        FINSObjMgr_enu
siebmtshmw 368      0       7      0.0   0.0   434496   0        FINSObjMgr_enu
siebmtshmw 3720     0       7      0.0   0.0   435468   0        FINSObjMgr_enu
siebmtshmw 3776     0       7      0.0   0.0   435628   0        FINSObjMgr_enu
siebmtshmw 380      0       7      0.0   0.0   436796   0        FINSObjMgr_enu
siebmtshmw 3864     0       7      0.0   0.0   433324   0        FINSObjMgr_enu
siebmtshmw 3928     0       7      0.0   0.0   430860   0        FINSObjMgr_enu
siebmtshmw 3984     0       7      0.0   0.0   433564   0        FINSObjMgr_enu
siebmtshmw 4028     0       7      0.0   0.0   435132   0        FINSObjMgr_enu
siebmtshmw 4060     0       7      0.0   0.0   432960   0        FINSObjMgr_enu
siebmtshmw 4092     0       7      0.0   0.0   433304   0        FINSObjMgr_enu
siebmtshmw 448      0       7      0.0   0.0   435912   0        FINSObjMgr_enu
siebmtshmw 640      0       7      0.0   0.0   432544   0        FINSObjMgr_enu
siebmtshmw 660      0       7      0.0   0.0   430724   0        FINSObjMgr_enu
siebmtshmw 692      0       7      0.0   0.0   436848   0        FINSObjMgr_enu
siebmtshmw 760      0       7      0.0   0.0   434668   0        FINSObjMgr_enu
siebmtsh   4068     0       25     0.0   0.0   11408    0        FSMSrvr
siebproc   764      0       1      0.0   0.0   8816     0        SCBroker
siebmtsh   2972     0       57     0.0   0.0   26844    0        SRBroker
siebmtsh   4076     0       8      0.0   0.0   97036    0        SRProc

SIEBEL_COMPONENT              CNT   THCNT  %CPU    TOT_MEM   AVG_MEM
================              ====  =====  ====    =======  ========
FINSObjMgr_enu                46    322     0.0   19987896  434519.5
FSMSrvr                       1     25      0.0      11408   11408.0
SCBroker                      1     1       0.0       8816    8816.0
SRBroker                      1     57      0.0      26844   26844.0
SRProc                        1     8       0.0      97036   97036.0
                              ====  =====  ====     ======  ========
           Cumulative         50    413       0   20132000
```

x460 System Single-Core Intel Xeon MP CPUs (Two Chassis, Single Partition)

Siebel Application Server Configuration

The following is the Korn shell script used to configure the Siebel Application Server in preparation for a 4,700 user run:

```
#!/bin/ksh
# Config Script Version Number: 1

function apply_config {
#
# Global Configuration Variables Used To Determine Parameters.
# Target No. Users: 4800
# No. Siebel App. Servers: 1
# No. Web Servers: 1
# App. Server CPU Bound?: False
# AOM SQL Cursor Cache: 0
# AOM SQL Data Cache: 0
# App. Server OSDF Latches Factor: 5280
# Web Client Threads: 4800
#

#
# Siebel Service Request Broker
#
COMPONENT=SRBroker
echo "change param MaxTasks=100 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MinMTServers=1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MaxMTServers=1 for comp "$COMPONENT >> $COMMAND_FILE

#
# Siebel Server Data Source
#
COMPONENT=ServerDataSrc
echo "change param DSMaxCachedCursors=0 for named subsystem "$COMPONENT >> $COMMAND_FILE
echo "change param DSMaxCachedDataSets=0 for named subsystem "$COMPONENT >> $COMMAND_FILE

#
# AOM Specific Configuration Variables Used To Determine Parameters.
# AOM1 Component Name: FINSObjMgr_enu
# No. Tasks per Thread: 1
# Thread Pooling Enabled?: False
# Thread Affinity Enabled?: True
# No. Users per DB Connection: 1
# Desired Users per AOM: 100
#
COMPONENT=FINSObjMgr_enu
echo "change param MaxTasks=4800 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MinMTServers=48 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MaxMTServers=48 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MinSharedDbConns=-1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MaxSharedDbConns=-1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MinTrxDbConns=-1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MinPoolThreads=0 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MaxPoolThreads=0 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param SessPerSisnConn=20 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param CommEnable=False for comp "$COMPONENT >> $COMMAND_FILE
echo "change param CommConfigManager=True for comp "$COMPONENT >> $COMMAND_FILE
echo "change param UseThreadPool=False for comp "$COMPONENT >> $COMMAND_FILE
echo "change param ThreadAffinity=True for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MemProtection=True for comp "$COMPONENT >> $COMMAND_FILE
echo "change EvtLogLvl EventContext=1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change EvtLogLvl ObjMgrSqlLog=1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change EvtLogLvl SQLFlags=0 for service "$COMPONENT >> $COMMAND_FILE

#
# AOM Specific Configuration Variables Used To Determine Parameters.
# AOM2 Component Name: eChannelObjMgr_enu
# No. Tasks per Thread: 1
# Thread Pooling Enabled?: False
# Thread Affinity Enabled?: True
# No. Users per DB Connection: 1
# Desired Users per AOM: 100
#
COMPONENT=eChannelObjMgr_enu
echo "change param MaxTasks=0 for comp "$COMPONENT >> $COMMAND_FILE
```

```
echo "change param MinMTServers=1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MaxMTServers=1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MinSharedDbConns=-1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MaxSharedDbConns=-1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MinTrxDbConns=-1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MinPoolThreads=0 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MaxPoolThreads=0 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param SessPerSisnConn=20 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param CommEnable=False for comp "$COMPONENT >> $COMMAND_FILE
echo "change param CommConfigManager=True for comp "$COMPONENT >> $COMMAND_FILE
echo "change param UseThreadPool=False for comp "$COMPONENT >> $COMMAND_FILE
echo "change param ThreadAffinity=True for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MemProtection=True for comp "$COMPONENT >> $COMMAND_FILE
echo "change EvtLogLvl EventContext=1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change EvtLogLvl ObjMgrSqlLog=1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change EvtLogLvl SQLFlags=0 for service "$COMPONENT >> $COMMAND_FILE

#
# AOM Specific Configuration Variables Used To Determine Parameters.
# AOM3 Component Name: WfProcMgr
# No. Tasks per Thread: 1
# Thread Pooling Enabled?: False
# Thread Affinity Enabled?: True
# No. Users per DB Connection: 2
# Desired Users per AOM: 100
#
COMPONENT=WfProcMgr
echo "change param MaxTasks=0 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MinMTServers=1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MaxMTServers=1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MinSharedDbConns=0 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MaxSharedDbConns=0 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MinTrxDbConns=0 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MinPoolThreads=0 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MaxPoolThreads=0 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param SessPerSisnConn=0 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param HonorMaxTasks=True for comp "$COMPONENT >> $COMMAND_FILE
echo "change param UsePrivatePort=True for comp "$COMPONENT >> $COMMAND_FILE
echo "change param BypassHandler=True for comp "$COMPONENT >> $COMMAND_FILE
echo "change param ModelCacheMax=84 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MemProtection=True for comp "$COMPONENT >> $COMMAND_FILE
echo "change EvtLogLvl EventContext=1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change EvtLogLvl ObjMgrSqlLog=1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change EvtLogLvl SQLFlags=0 for service "$COMPONENT >> $COMMAND_FILE

#
# Apply the configuration changes to the Siebel Application Server
#
"c:/sea77/siebsrvr/bin/srvrmgr" $CONNECT_STRING <<EOF
read $COMMAND_FILE
quit
EOF

printf "\n'srvrmgr' Connect String Used: %s\n" "$CONNECT_STRING"
printf "Application Server Configured: %s\n\n" "$COMPUTERNAME"
}

################################################################################
################################################################################
#
# Main Script Starts Here
#
################################################################################
################################################################################

CONNECT_STRING="-e siebel -g $COMPUTERNAME -u sadmin -p sadmin"
COMMAND_FILE=c:/temp/srvrmgr.$$
OUTPUT_FILE=c:/temp/srvrmgr.out.$$

if [[ -f $COMMAND_FILE ]]; then
   rm $COMMAND_FILE
fi

if [[ -f $OUTPUT_FILE ]]; then
   rm $OUTPUT_FILE
fi

#
# Specify the App Server we want to configure
#
```

```
echo "set server "$COMPUTERNAME >> $COMMAND_FILE

#
# Execute the 'set server' command in 'srvrmgr' and capture the OUTPUT
#
"c:/sea77/siebsrvr/bin/srvrmgr" $CONNECT_STRING <<EOF >$OUTPUT_FILE
read $COMMAND_FILE
quit
EOF

#
# GREP the OUTPUT from the 'srvrmgr' command for our App Server Name in the
# PROMPT. The PROMPT will take the format 'srvrmgr:<App Server Name>', if
# the 'set server' command was successful
#
if [[ -z "`grep \"srvrmgr:$COMPUTERNAME\" $OUTPUT_FILE`" ]]; then
   printf "\n\nERROR! - Siebel App Server '%s' Not Found In Enterprise '%s'!\n\n"
"$COMPUTERNAME"  "siebel"
else
   #
   # App Server exists! Apply the config to it. The FUNCTION continues to APPEND
   # the config details to the COMMAND_FILE, which at the moment, contains just
   # the valid 'set server' command
   #
   apply_config
fi

rm $COMMAND_FILE
rm $OUTPUT_FILE
```

Siebel Run-time Processes

The following is a custom display of the Siebel Application Server processes that were running before the start of the 4,700 user run:

```
COMMAND    PID      SC     THCNT  %CPU  %MEM  VSZ      TASK_ID  SIEBEL_COMPONENT
siebmtshmw 10196    0      6      0.0   0.0   61844    0        FINSObjMgr_enu
siebmtshmw 11328    0      6      0.0   0.0   61840    0        FINSObjMgr_enu
siebmtshmw 11760    0      6      0.0   0.0   61840    0        FINSObjMgr_enu
siebmtshmw 12584    0      6      0.0   0.0   61848    0        FINSObjMgr_enu
siebmtshmw 13260    0      6      0.0   0.0   61840    0        FINSObjMgr_enu
siebmtshmw 13424    0      6      0.0   0.0   61844    0        FINSObjMgr_enu
siebmtshmw 13540    0      6      0.0   0.0   61844    0        FINSObjMgr_enu
siebmtshmw 13732    0      6      0.0   0.0   61844    0        FINSObjMgr_enu
siebmtshmw 13804    0      6      0.0   0.0   61844    0        FINSObjMgr_enu
siebmtshmw 13912    0      6      0.0   0.0   61844    0        FINSObjMgr_enu
siebmtshmw 14036    0      6      0.0   0.0   61836    0        FINSObjMgr_enu
siebmtshmw 14564    0      6      0.0   0.0   61848    0        FINSObjMgr_enu
siebmtshmw 14588    0      6      0.0   0.0   61840    0        FINSObjMgr_enu
siebmtshmw 14952    0      6      0.0   0.0   61844    0        FINSObjMgr_enu
siebmtshmw 15020    0      6      0.0   0.0   61844    0        FINSObjMgr_enu
siebmtshmw 15144    0      6      0.0   0.0   61840    0        FINSObjMgr_enu
siebmtshmw 15964    0      6      0.0   0.0   61844    0        FINSObjMgr_enu
siebmtshmw 16132    0      6      0.0   0.0   61840    0        FINSObjMgr_enu
siebmtshmw 17184    0      6      0.0   0.0   61840    0        FINSObjMgr_enu
siebmtshmw 17368    0      6      0.0   0.0   61844    0        FINSObjMgr_enu
siebmtshmw 17812    0      7      0.0   0.0   61852    0        FINSObjMgr_enu
siebmtshmw 18108    0      6      0.0   0.0   61848    0        FINSObjMgr_enu
siebmtshmw 18484    0      6      0.0   0.0   61840    0        FINSObjMgr_enu
siebmtshmw 20092    0      6      0.0   0.0   61852    0        FINSObjMgr_enu
siebmtshmw 20152    0      6      0.0   0.0   61840    0        FINSObjMgr_enu
siebmtshmw 20276    0      6      0.0   0.0   61840    0        FINSObjMgr_enu
siebmtshmw 20668    0      6      0.0   0.0   61840    0        FINSObjMgr_enu
siebmtshmw 20672    0      6      0.0   0.0   61836    0        FINSObjMgr_enu
siebmtshmw 20872    0      6      0.0   0.0   61840    0        FINSObjMgr_enu
siebmtshmw 2100     0      6      0.0   0.0   61828    0        FINSObjMgr_enu
siebmtshmw 2164     0      6      0.0   0.0   61844    0        FINSObjMgr_enu
siebmtshmw 22840    0      6      0.0   0.0   61844    0        FINSObjMgr_enu
siebmtshmw 22852    0      7      0.0   0.0   61852    0        FINSObjMgr_enu
siebmtshmw 22876    0      7      0.0   0.0   61852    0        FINSObjMgr_enu
siebmtshmw 22968    0      6      0.0   0.0   61840    0        FINSObjMgr_enu
siebmtshmw 23804    0      6      0.0   0.0   61836    0        FINSObjMgr_enu
siebmtshmw 4008     0      6      0.0   0.0   61848    0        FINSObjMgr_enu
siebmtshmw 4132     0      6      0.0   0.0   61844    0        FINSObjMgr_enu
siebmtshmw 4692     0      6      0.0   0.0   61840    0        FINSObjMgr_enu
siebmtshmw 4980     0      7      0.0   0.0   61856    0        FINSObjMgr_enu
siebmtshmw 6120     0      6      0.0   0.0   61844    0        FINSObjMgr_enu
siebmtshmw 6868     0      6      0.0   0.0   61840    0        FINSObjMgr_enu
siebmtshmw 7744     0      7      0.0   0.0   61848    0        FINSObjMgr_enu
siebmtshmw 7760     0      6      0.0   0.0   61836    0        FINSObjMgr_enu
siebmtshmw 8320     0      6      0.0   0.0   61844    0        FINSObjMgr_enu
siebmtshmw 8336     0      6      0.0   0.0   61844    0        FINSObjMgr_enu
siebmtshmw 8340     0      6      0.0   0.0   61844    0        FINSObjMgr_enu
siebmtshmw 9928     0      6      0.0   0.0   61840    0        FINSObjMgr_enu
siebmtsh   23916    0      25     0.0   0.0   11420    0        FSMSrvr
siebproc   23296    0      1      0.0   0.0   8456     0        SCBroker
siebmtsh   13772    0      16     0.0   0.0   20736    0        SRBroker
siebmtsh   19112    0      8      0.0   0.0   97104    0        SRProc

SIEBEL_COMPONENT              CNT    THCNT  %CPU    TOT_MEM    AVG_MEM
================              ====   =====  ====    =======   ========
FINSObjMgr_enu                48     293     0.0    2968464    61843.0
FSMSrvr                       1      25      0.0      11420    11420.0
SCBroker                      1      1       0.0       8456     8456.0
SRBroker                      1      16      0.0      20736    20736.0
SRProc                        1      8       0.0      97104    97104.0
                              ====   =====  ====     ======   ========
           Cumulative         52     343       0    3106180
```

The following are the Siebel Application Server processes that were running after the 4,700 user run:

```
COMMAND    PID      SC     THCNT  %CPU  %MEM  VSZ      TASK_ID  SIEBEL_COMPONENT
siebmtshmw 10196    0      12     0.0   0.0   440908   0        FINSObjMgr_enu
siebmtshmw 11328    0      12     0.0   0.0   443276   0        FINSObjMgr_enu
siebmtshmw 11760    0      12     0.0   0.0   440392   0        FINSObjMgr_enu
siebmtshmw 12584    0      12     0.0   0.0   440892   0        FINSObjMgr_enu
siebmtshmw 13260    0      12     0.0   0.0   441016   0        FINSObjMgr_enu
siebmtshmw 13424    0      12     0.0   0.0   437192   0        FINSObjMgr_enu
siebmtshmw 13540    0      12     0.0   0.0   441036   0        FINSObjMgr_enu
siebmtshmw 13732    0      12     0.0   0.0   440972   0        FINSObjMgr_enu
siebmtshmw 13804    0      12     0.0   0.0   438672   0        FINSObjMgr_enu
siebmtshmw 13912    0      12     0.0   0.0   442620   0        FINSObjMgr_enu
siebmtshmw 14036    0      12     0.0   0.0   444020   0        FINSObjMgr_enu
siebmtshmw 14564    0      12     0.0   0.0   439864   0        FINSObjMgr_enu
siebmtshmw 14588    0      12     0.0   0.0   437384   0        FINSObjMgr_enu
siebmtshmw 14952    0      12     0.0   0.0   440420   0        FINSObjMgr_enu
siebmtshmw 15020    0      12     0.0   0.0   439584   0        FINSObjMgr_enu
siebmtshmw 15144    0      12     0.0   0.0   440412   0        FINSObjMgr_enu
siebmtshmw 15964    0      12     0.0   0.0   437804   0        FINSObjMgr_enu
siebmtshmw 16132    0      12     0.0   0.0   438252   0        FINSObjMgr_enu
siebmtshmw 17184    0      12     0.0   0.0   441236   0        FINSObjMgr_enu
siebmtshmw 17368    0      12     0.0   0.0   440708   0        FINSObjMgr_enu
siebmtshmw 17812    0      12     0.0   0.0   441628   0        FINSObjMgr_enu
siebmtshmw 18108    0      12     0.0   0.0   440820   0        FINSObjMgr_enu
siebmtshmw 18484    0      12     0.0   0.0   441356   0        FINSObjMgr_enu
siebmtshmw 20092    0      12     0.0   0.0   438544   0        FINSObjMgr_enu
siebmtshmw 20152    0      12     0.0   0.0   440612   0        FINSObjMgr_enu
siebmtshmw 20276    0      12     0.0   0.0   443576   0        FINSObjMgr_enu
siebmtshmw 20668    0      12     0.0   0.0   440484   0        FINSObjMgr_enu
siebmtshmw 20672    0      12     0.0   0.0   438544   0        FINSObjMgr_enu
siebmtshmw 20872    0      12     0.0   0.0   444076   0        FINSObjMgr_enu
siebmtshmw 2100     0      12     0.0   0.0   437160   0        FINSObjMgr_enu
siebmtshmw 2164     0      12     0.0   0.0   442176   0        FINSObjMgr_enu
siebmtshmw 22840    0      12     0.0   0.0   441924   0        FINSObjMgr_enu
siebmtshmw 22852    0      12     0.0   0.0   440436   0        FINSObjMgr_enu
siebmtshmw 22876    0      12     0.0   0.0   439756   0        FINSObjMgr_enu
siebmtshmw 22968    0      12     0.0   0.0   438744   0        FINSObjMgr_enu
siebmtshmw 23804    0      12     0.0   0.0   440896   0        FINSObjMgr_enu
siebmtshmw 4008     0      12     0.0   0.0   441552   0        FINSObjMgr_enu
siebmtshmw 4132     0      12     0.0   0.0   439768   0        FINSObjMgr_enu
siebmtshmw 4692     0      12     0.0   0.0   440140   0        FINSObjMgr_enu
siebmtshmw 4980     0      12     0.0   0.0   442112   0        FINSObjMgr_enu
siebmtshmw 6120     0      12     0.0   0.0   438916   0        FINSObjMgr_enu
siebmtshmw 6868     0      12     0.0   0.0   438376   0        FINSObjMgr_enu
siebmtshmw 7744     0      12     0.0   0.0   442452   0        FINSObjMgr_enu
siebmtshmw 7760     0      12     0.0   0.0   441500   0        FINSObjMgr_enu
siebmtshmw 8320     0      12     0.0   0.0   437964   0        FINSObjMgr_enu
siebmtshmw 8336     0      12     0.0   0.0   442828   0        FINSObjMgr_enu
siebmtshmw 8340     0      12     0.0   0.0   438812   0        FINSObjMgr_enu
siebmtshmw 9928     0      12     0.0   0.0   439504   0        FINSObjMgr_enu
siebmtsh   23916    0      25     0.0   0.0   11420    0        FSMSrvr
siebproc   23296    0      1      0.0   0.0   8896     0        SCBroker
siebmtsh   13772    0      59     0.0   0.0   27224    0        SRBroker
siebmtsh   19112    0      8      0.0   0.0   97104    0        SRProc

SIEBEL_COMPONENT              CNT    THCNT  %CPU    TOT_MEM    AVG_MEM
================              ====   =====  ====    =======  ========
FINSObjMgr_enu                48     576     0.0   21141316  440444.1
FSMSrvr                       1      25      0.0      11420   11420.0
SCBroker                      1      1       0.0       8896    8896.0
SRBroker                      1      59      0.0      27224   27224.0
SRProc                        1      8       0.0      97104   97104.0
                              ====   =====  ====     ======  ========
           Cumulative         52     669       0   21285960
```

Appendix E: Siebel Intel Performance Study Summary Results

In Table 4 through Table 6, the results have been color-coded to allow easier identification of each result set on the graphs in Graph 3 and Graph 4 that follow at the end of the appendix.

Intel P-IV: Single-Core Intel Xeon MP CPUs

In 2005, Siebel ran a performance study of Siebel 7.7 application software using the best Intel 4-way white-box server configurations that were available at the time. Three of the runs were completed with servers installed with Intel Xeon MP CPUs.

With permission from Siebel, the summary data from these runs has been reproduced in Table 4 through Table 6.

4x 2.8 GHz CPUs with 2MB L3 Cache and Windows Hyper-Threading OFF

Users	No. of CPUs	No. of CPU Cores	Total CPU%	User CPU%	System CPU%	Hits/Sec	Avg Response	Users/CPU Core (@100%)	Users/CPU Core (actual)
400	4	4	25.83	23.67	2.17	50.00	0.09	387	100
500	4	4	32.68	30.19	2.49	63.00	0.09	382	125
600	4	4	39.64	36.69	2.95	75.00	0.09	378	150
700	4	4	46.70	43.30	3.42	87.00	0.09	375	175
800	4	4	54.30	50.36	3.94	100.00	0.10	368	200
900	4	4	62.16	57.60	4.56	112.00	0.10	362	225
1000	4	4	69.50	64.50	5.02	125.00	0.11	360	250
1100	4	4	77.85	72.12	5.73	138.00	0.12	353	275
1200	4	4	86.20	80.10	6.10	150.00	0.15	348	300
1300	4	4	93.03	86.60	6.46	161.00	0.21	349	325

Table 4: 2MB L3 Cache with Windows Hyper-Threading OFF

4x 2.8 GHz CPUs with 2MB L3 Cache and Windows Hyper-Threading ON

Users	No. of CPUs	No. of CPU Cores	Total CPU%	User CPU%	System CPU%	Hits/Sec	Avg Response	Users/CPU Core (@100%)	Users/CPU Core (actual)
500	4	4	18.42	16.93	1.49	63.00	0.10	679	125
600	4	4	22.77	21.19	1.58	75.00	0.10	659	150
700	4	4	27.77	25.70	2.00	87.00	0.10	630	175
800	4	4	33.00	30.70	2.31	100.00	0.10	606	200
900	4	4	38.60	35.89	2.81	112.00	0.10	583	225
1000	4	4	45.93	42.78	3.15	125.00	0.11	544	250
1100	4	4	53.55	49.85	3.70	137.00	0.12	514	275
1200	4	4	62.17	57.94	4.23	150.00	0.13	483	300
1300	4	4	70.47	65.90	4.57	162.00	0.14	461	325
1400	4	4	81.22	75.93	5.38	174.00	0.17	431	350
1500	4	4	93.23	87.06	6.17	184.00	0.33	402	375

Table 5: 2MB L3 Cache with Windows Hyper-Threding ON

4x 2.8 GHz CPUs with 4MB L3 Cache and Windows Hyper-Threading ON

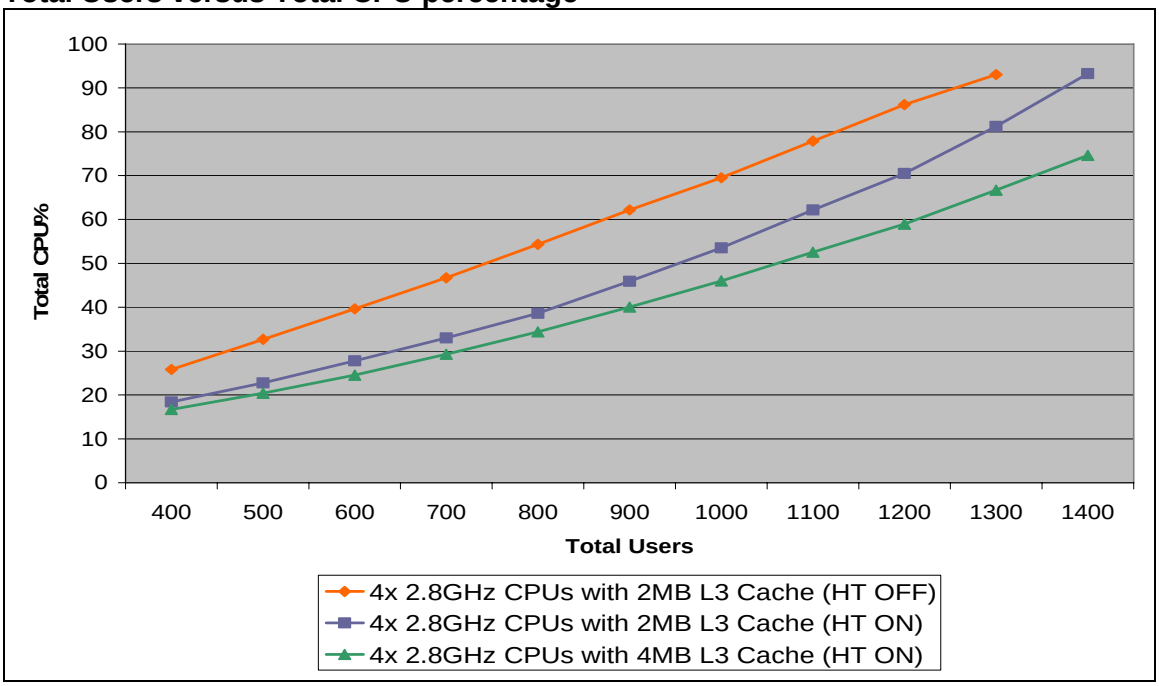
Users	No. of CPUs	No. of CPU Cores	Total CPU%	User CPU%	System CPU%	Hits/Sec	Avg Response	Users/CPU Core (@100%)	Users/CPU Core (actual)
500	4	4	16.70	15.48	1.21	63.00	0.10	749	125
600	4	4	20.41	19.00	1.42	75.00	0.10	735	150
700	4	4	24.53	22.89	1.66	88.00	0.10	713	175
800	4	4	29.27	27.35	1.92	100.00	0.10	683	200
900	4	4	34.38	32.05	2.33	113.00	0.10	654	225
1000	4	4	40.03	37.33	2.69	125.00	0.11	625	250
1100	4	4	45.99	42.89	3.10	138.00	0.12	598	275
1200	4	4	52.54	49.09	3.45	150.00	0.12	571	300
1300	4	4	58.96	55.22	3.73	162.00	0.12	551	325
1400	4	4	66.67	62.47	4.20	175.00	0.14	525	350
1500	4	4	74.61	70.00	4.61	187.00	0.14	503	375

Table 6: 4MB L3 Cache with Windows Hyper-Threading ON

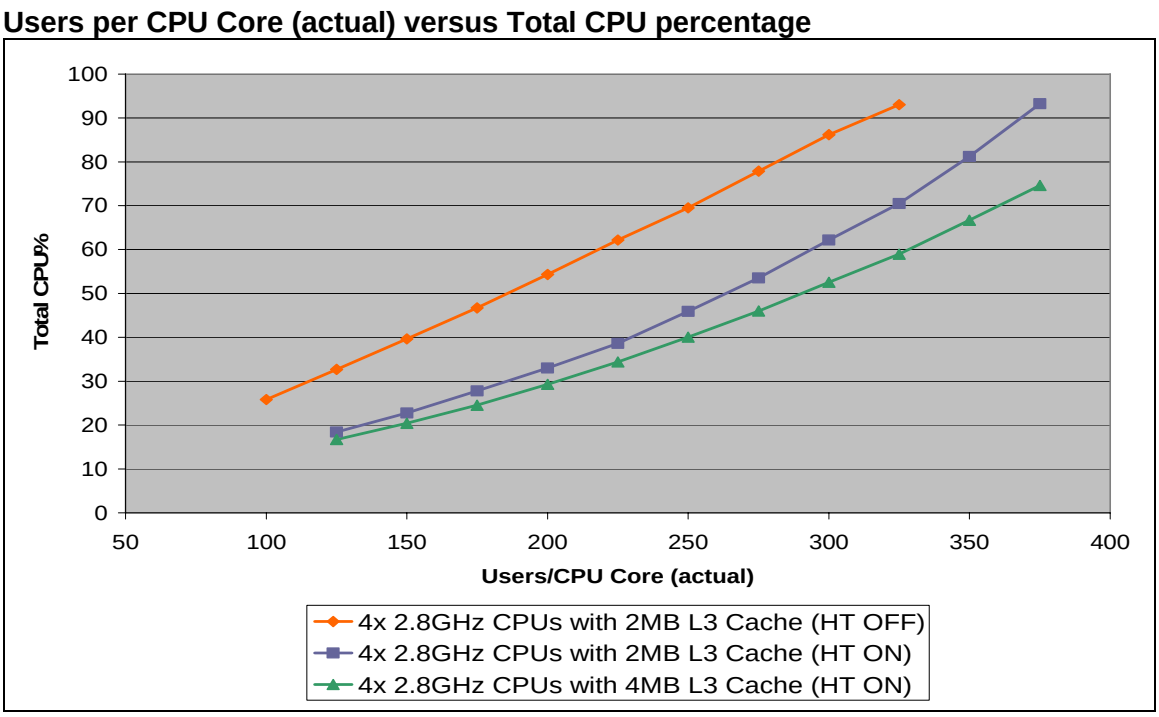
Graphical Representations of Summary Results

The graphs in Graph 3 and Graph 4 show the summary results.

Total Users versus Total CPU percentage



Graph 3: Total Users versus Total CPU percentage



Graph 4: Users per CPU Core (actual) versus Total CPU percentage

Appendix F: PSPP Benchmark Summary Results

In Table 7 through Table 10, the results have been color-coded to allow easier identification of each result set on the graphs in Graph 5 and Graph 6 that follow at the end of the appendix.

x366 System Single-Core Intel Xeon MP CPUs

The data in Table 7 was captured for the x366 system fitted with 4x single-core, 3.6 GHz Intel Xeon MP CPUs. This was a single-chassis configuration which had 4x 3.6 GHz CPU Cores.

Users	No. of CPUs	No. of CPU Cores	Steady State Total CPU (%)	Users/CPU Core (@100%)	Users/CPU Core (actual)	Steady State Free RAM (MB)	Total Installed RAM (MB)	RAM Used (MB)	RAM/User (MB)
2,100	4	4	83.4	629	525	12,768	24,576	11,808	5.62

Table 7: x366 System Single-Core Intel Xeon MP CPUs

x366 System Dual-Core Intel Xeon MP CPUs

The data in Table 8 was captured for the x366 system fitted with 4x dual-core, 3.0 GHz Intel Xeon MP CPUs. This was a single-chassis configuration which had 8x 3.0 GHz CPU Cores.

Users	No. of CPUs	No. of CPU Cores	Steady State Total CPU (%)	Users/CPU Core (@100%)	Users/CPU Core (actual)	Steady State Free RAM (MB)	Total Installed RAM (MB)	RAM Used (MB)	RAM/User (MB)
3,600	4	8	84.7	531	450	5,158	24,576	19,418	5.39

Table 8: x366 System Dual-Core Intel Xeon MP CPUs

x460 System Single-Core Intel Xeon MP CPUs (One Chassis)

The data in Table 9 was captured for the x460 system fitted with 4x single-core, 3.3 GHz Intel Xeon MP CPUs. This was a single-chassis configuration which had 4x 3.3 GHz CPU Cores (each chassis has 4x single-core CPUs).

Users	No. of CPUs	No. of CPU Cores	Steady State Total CPU (%)	Users/CPU Core (@100%)	Users/CPU Core (actual)	Steady State Free RAM (MB)	Total Installed RAM (MB)	RAM Used (MB)	RAM/User (MB)
2,000	4	4	84.0	596	500	5,192	16,384	11,192	5.60

Table 9: x460 System Single-Core Intel Xeon MP CPUs (One Chassis)

x460 System Single-Core Intel Xeon MP CPUs (Two Chassis, Single Partition)

The data in Table 10 was captured for the x460 system fitted with 8x single-core, 3.3 GHz Intel Xeon MP CPUs. This was a two-chassis, single-partition configuration which had 8x 3.3 GHz CPU Cores (each chassis has 4x single-core CPUs).

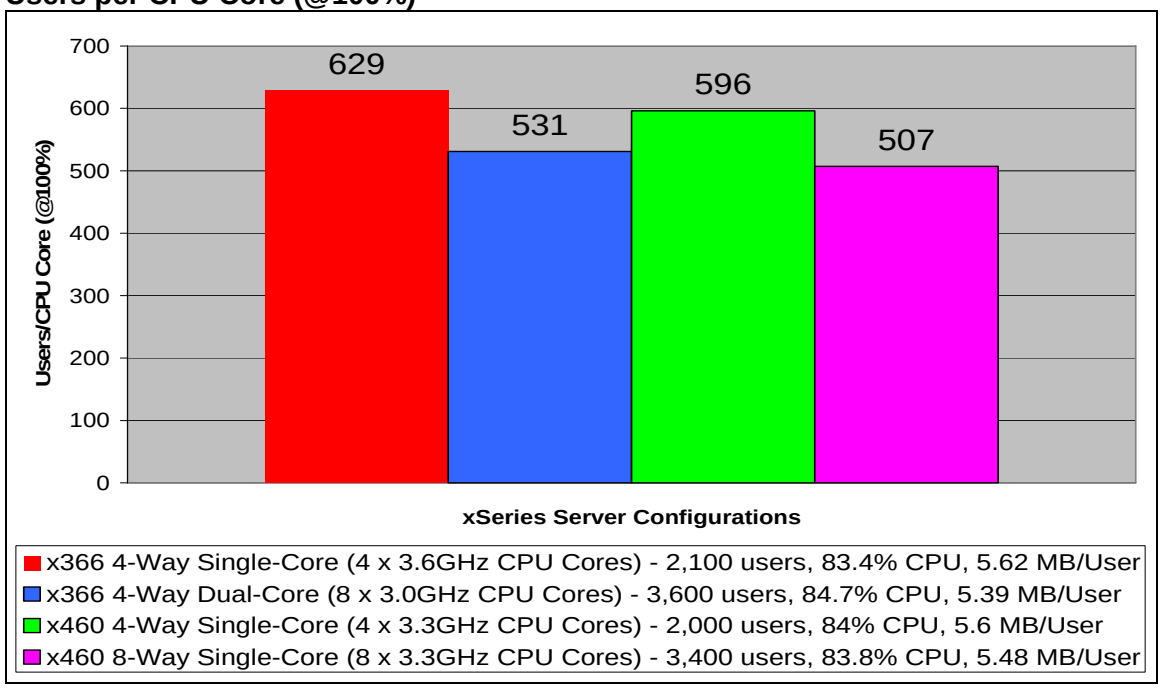
Users	No. of CPUs	No. of CPU Cores	Steady State Total CPU (%)	Users/CPU Core (@100%)	Users/CPU Core (actual)	Steady State Free RAM (MB)	Total Installed RAM (MB)	RAM Used (MB)	RAM/User (MB)
3,400	8	8	83.8	507	425	13,612	32,256	18,644	5.48

Table 10: x460 System Single-Core Intel Xeon MP CPUs (Two Chassis, Single Partition)

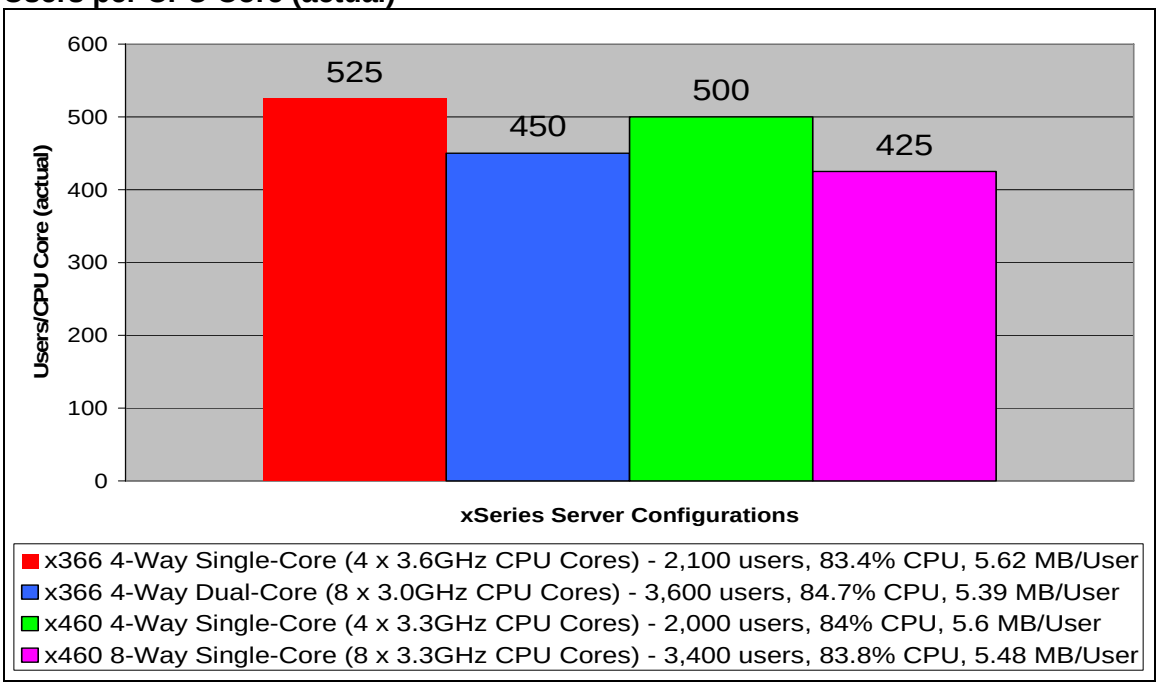
Graphical Representations of Summary Results

The graphs in Graph 5 and Graph 6 show the summary results.

Users per CPU Core (@100%)



Graph 5: All xSeries Systems



Graph 6: All xSeries Systems (actual)

Appendix G: PSPP Benchmark Siebel Application Server Configuration Examples

Before the best PSPP benchmark run was captured, the Siebel 7.7 Application Server was reconfigured to support slightly more Siebel users than targeted by the LoadRunner scenario. As a rule of thumb, the application server was always configured for 100 to 200 more users than the scenario called for.

Many test runs were performed to determine the best Siebel component base configuration to use. Once determined, all future configurations were scaled up versions of the base, using the same option settings and parameter ratios.

The following sections illustrate the Siebel Application Server configurations that were applied to achieve the best run for each xSeries hardware category.

x366 System Single-Core Intel Xeon MP CPUs

Siebel Application Server Configuration

The following is the Korn shell script used to configure the Siebel Application Server in preparation for a 2,100 user run:

```
#!/bin/ksh
# Config Script Version Number: 1

function apply_config {
#
# Global Configuration Variables Used To Determine Parameters.
# Target No. Users: 2200
# No. Siebel App. Servers: 1
# No. Web Servers: 1
# App. Server CPU Bound?: False
# AOM SQL Cursor Cache: 0
# AOM SQL Data Cache: 0
# App. Server OSDF Latches Factor: 2517
# Web Client Threads: 2200
#


#
# Siebel Service Request Broker
#
COMPONENT=SRBroker
echo "change param MaxTasks=100 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MinMTServers=1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MaxMTServers=1 for comp "$COMPONENT >> $COMMAND_FILE

#
# Siebel Server Data Source
#
COMPONENT=ServerDataSrc
echo "change param DSMaxCachedCursors=0 for named subsystem "$COMPONENT >> $COMMAND_FILE
echo "change param DSMaxCachedDataSets=0 for named subsystem "$COMPONENT >> $COMMAND_FILE

#
# AOM Specific Configuration Variables Used To Determine Parameters.
# AOM1 Component Name: FINSObjMgr_enu
# No. Tasks per Thread: 1
# Thread Pooling Enabled?: False
# Thread Affinity Enabled?: True
# No. Users per DB Connection: 1
# Desired Users per AOM: 100
#
COMPONENT=FINSObjMgr_enu
echo "change param MaxTasks=1760 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MinMTServers=18 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MaxMTServers=18 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MinSharedDbConns=-1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MaxSharedDbConns=-1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MinTrxDbConns=-1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MinPoolThreads=0 for comp "$COMPONENT >> $COMMAND_FILE
```

```
echo "change param MaxPoolThreads=0 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param SessPerSisnConn=20 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param CommEnable=False for comp "$COMPONENT >> $COMMAND_FILE
echo "change param CommConfigManager=True for comp "$COMPONENT >> $COMMAND_FILE
echo "change param UseThreadPool=False for comp "$COMPONENT >> $COMMAND_FILE
echo "change param ThreadAffinity=True for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MemProtection=True for comp "$COMPONENT >> $COMMAND_FILE
echo "change EvtLogLvl EventContext=1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change EvtLogLvl ObjMgrSqlLog=1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change EvtLogLvl SQLFlags=0 for service "$COMPONENT >> $COMMAND_FILE

#
# AOM Specific Configuration Variables Used To Determine Parameters.
# AOM2 Component Name: eChannelObjMgr_enu
# No. Tasks per Thread: 1
# Thread Pooling Enabled?: False
# Thread Affinity Enabled?: True
# No. Users per DB Connection: 1
# Desired Users per AOM: 100
#
COMPONENT=eChannelObjMgr_enu
echo "change param MaxTasks=440 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MinMTServers=4 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MaxMTServers=4 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MinSharedDbConns=-1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MaxSharedDbConns=-1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MinTrxDbConns=-1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MinPoolThreads=0 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MaxPoolThreads=0 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param SessPerSisnConn=20 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param CommEnable=False for comp "$COMPONENT >> $COMMAND_FILE
echo "change param CommConfigManager=True for comp "$COMPONENT >> $COMMAND_FILE
echo "change param UseThreadPool=False for comp "$COMPONENT >> $COMMAND_FILE
echo "change param ThreadAffinity=True for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MemProtection=True for comp "$COMPONENT >> $COMMAND_FILE
echo "change EvtLogLvl EventContext=1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change EvtLogLvl ObjMgrSqlLog=1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change EvtLogLvl SQLFlags=0 for service "$COMPONENT >> $COMMAND_FILE

#
# AOM Specific Configuration Variables Used To Determine Parameters.
# AOM3 Component Name: WfProcMgr
# No. Tasks per Thread: 1
# Thread Pooling Enabled?: False
# Thread Affinity Enabled?: True
# No. Users per DB Connection: 2
# Desired Users per AOM: 100
#
COMPONENT=WfProcMgr
echo "change param MaxTasks=88 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MinMTServers=1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MaxMTServers=1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MinSharedDbConns=44 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MaxSharedDbConns=44 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MinTrxDbConns=44 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MinPoolThreads=0 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MaxPoolThreads=0 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param SessPerSisnConn=0 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param HonorMaxTasks=True for comp "$COMPONENT >> $COMMAND_FILE
echo "change param UsePrivatePort=True for comp "$COMPONENT >> $COMMAND_FILE
echo "change param BypassHandler=True for comp "$COMPONENT >> $COMMAND_FILE
echo "change param ModelCacheMax=84 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MemProtection=True for comp "$COMPONENT >> $COMMAND_FILE
echo "change EvtLogLvl EventContext=1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change EvtLogLvl ObjMgrSqlLog=1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change EvtLogLvl SQLFlags=0 for service "$COMPONENT >> $COMMAND_FILE

#
# Apply the configuration changes to the Siebel Application Server
#
"c:/sea77/siebsrvr/bin/srvrmgr" $CONNECT_STRING <<EOF
read $COMMAND_FILE
quit
EOF

printf "\n'srvrmgr' Connect String Used: %s\n" "$CONNECT_STRING"
printf "Application Server Configured: %s\n\n" "$COMPUTERNAME"
}

################################################################################
```

```
################################################################################
#
# Main Script Starts Here
#
################################################################################
################################################################################

CONNECT_STRING="-e siebel -g $COMPUTERNAME -u sadmin -p sadmin"
COMMAND_FILE=c:/temp/srvrmgr.$$
OUTPUT_FILE=c:/temp/srvrmgr.out.$$

if [[ -f $COMMAND_FILE ]]; then
   rm $COMMAND_FILE
fi

if [[ -f $OUTPUT_FILE ]]; then
   rm $OUTPUT_FILE
fi

#
# Specify the App Server we want to configure
#
echo "set server "$COMPUTERNAME >> $COMMAND_FILE

#
# Execute the 'set server' command in 'srvrmgr' and capture the OUTPUT
#
"c:/sea77/siebsrvr/bin/srvrmgr" $CONNECT_STRING <<EOF >$OUTPUT_FILE
read $COMMAND_FILE
quit
EOF

#
# GREP the OUTPUT from the 'srvrmgr' command for our App Server Name in the
# PROMPT. The PROMPT will take the format 'srvrmgr:<App Server Name>', if
# the 'set server' command was successful
#
if [[ -z "`grep \"srvrmgr:$COMPUTERNAME\" $OUTPUT_FILE`" ]]; then
   printf "\n\nERROR! - Siebel App Server '%s' Not Found In Enterprise '%s'!\n\n"
"$COMPUTERNAME"  "siebel"
else
   #
   # App Server exists! Apply the config to it. The FUNCTION continues to APPEND
   # the config details to the COMMAND_FILE, which at the moment, contains just
   # the valid 'set server' command
   #
   apply_config
fi

rm $COMMAND_FILE
rm $OUTPUT_FILE
```

Siebel Run-time Processes

The following is a custom display of the Siebel Application Server processes that were running before the start of the 2,100 user run:

```
COMMAND    PID      SC     THCNT  %CPU  %MEM  VSZ      TASK_ID  SIEBEL_COMPONENT
siebmtshmw 1256     0      6      0.0   0.0   35028    0        FINSObjMgr_enu
siebmtshmw 136      0      6      0.0   0.0   35024    0        FINSObjMgr_enu
siebmtshmw 1368     0      7      0.0   0.0   35032    0        FINSObjMgr_enu
siebmtshmw 1572     0      6      0.0   0.0   35020    0        FINSObjMgr_enu
siebmtshmw 1740     0      7      0.0   0.0   35016    0        FINSObjMgr_enu
siebmtshmw 1744     0      6      0.0   0.0   35028    0        FINSObjMgr_enu
siebmtshmw 1760     0      6      0.0   0.0   35024    0        FINSObjMgr_enu
siebmtshmw 1880     0      6      0.0   0.0   35024    0        FINSObjMgr_enu
siebmtshmw 1900     0      7      0.0   0.0   35032    0        FINSObjMgr_enu
siebmtshmw 2096     0      6      0.0   0.0   35024    0        FINSObjMgr_enu
siebmtshmw 2100     0      6      0.0   0.0   35020    0        FINSObjMgr_enu
siebmtshmw 2160     0      6      0.0   0.0   35024    0        FINSObjMgr_enu
siebmtshmw 2168     0      7      0.0   0.0   35020    0        FINSObjMgr_enu
siebmtshmw 2184     0      6      0.0   0.0   35020    0        FINSObjMgr_enu
siebmtshmw 2472     0      6      0.0   0.0   35020    0        FINSObjMgr_enu
siebmtshmw 3604     0      6      0.0   0.0   35028    0        FINSObjMgr_enu
siebmtshmw 3752     0      6      0.0   0.0   35024    0        FINSObjMgr_enu
siebmtshmw 3768     0      6      0.0   0.0   35024    0        FINSObjMgr_enu
siebmtsh   3572     0      25     0.0   0.0   11300    0        FSMSrvr
siebproc   3748     0      1      0.0   0.0   7904     0        SCBroker
siebmtsh   1216     0      17     0.0   0.0   20176    0        SRBroker
siebmtsh   3504     0      8      0.0   0.0   96352    0        SRProc
siebmtshmw 1604     0      94     0.0   0.0   142324   0        WfProcMgr
siebmtshmw 1940     0      6      0.0   0.0   23620    0        eChannelObjMgr_enu
siebmtshmw 2544     0      6      0.0   0.0   23612    0        eChannelObjMgr_enu
siebmtshmw 3208     0      6      0.0   0.0   23624    0        eChannelObjMgr_enu
siebmtshmw 3612     0      7      0.0   0.0   23640    0        eChannelObjMgr_enu

SIEBEL_COMPONENT              CNT    THCNT  %CPU    TOT_MEM   AVG_MEM
================              ====   =====  ====    =======   ========
FINSObjMgr_enu                18     112     0.0     630432   35024.0
FSMSrvr                       1      25      0.0      11300   11300.0
SCBroker                      1      1       0.0       7904    7904.0
SRBroker                      1      17      0.0      20176   20176.0
SRProc                        1      8       0.0      96352   96352.0
WfProcMgr                     1      94      0.0     142324  142324.0
eChannelObjMgr_enu            4      25      0.0      94496   23624.0
                              ====   =====  ====     ======  ========
           Cumulative         27     282       0    1002984
```

The following are the Siebel Application Server processes that were running after the 2,100 user run:

```
COMMAND    PID      SC      THCNT  %CPU  %MEM  VSZ      TASK_ID  SIEBEL_COMPONENT
siebmtshmw 1256     0       13     0.0   0.0   396064   0        FINSObjMgr_enu
siebmtshmw 136      0       13     0.0   0.0   393984   0        FINSObjMgr_enu
siebmtshmw 1368     0       13     0.0   0.0   396576   0        FINSObjMgr_enu
siebmtshmw 1572     0       13     0.0   0.0   393620   0        FINSObjMgr_enu
siebmtshmw 1740     0       13     0.0   0.0   396436   0        FINSObjMgr_enu
siebmtshmw 1744     0       13     0.0   0.0   394252   0        FINSObjMgr_enu
siebmtshmw 1760     0       13     0.0   0.0   394488   0        FINSObjMgr_enu
siebmtshmw 1880     0       13     0.0   0.0   397212   0        FINSObjMgr_enu
siebmtshmw 1900     0       13     0.0   0.0   393028   0        FINSObjMgr_enu
siebmtshmw 2096     0       13     0.0   0.0   392848   0        FINSObjMgr_enu
siebmtshmw 2100     0       13     0.0   0.0   393064   0        FINSObjMgr_enu
siebmtshmw 2160     0       13     0.0   0.0   399436   0        FINSObjMgr_enu
siebmtshmw 2168     0       13     0.0   0.0   397880   0        FINSObjMgr_enu
siebmtshmw 2184     0       13     0.0   0.0   392588   0        FINSObjMgr_enu
siebmtshmw 2472     0       13     0.0   0.0   395012   0        FINSObjMgr_enu
siebmtshmw 3604     0       13     0.0   0.0   393196   0        FINSObjMgr_enu
siebmtshmw 3752     0       13     0.0   0.0   396668   0        FINSObjMgr_enu
siebmtshmw 3768     0       13     0.0   0.0   392224   0        FINSObjMgr_enu
siebmtsh   3572     0       25     0.0   0.0   11300    0        FSMSrvr
siebproc   3748     0       1      0.0   0.0   8212     0        SCBroker
siebmtsh   1216     0       37     0.0   0.0   23372    0        SRBroker
siebmtsh   3504     0       8      0.0   0.0   96376    0        SRProc
siebmtshmw 1604     0       95     0.0   0.0   219500   0        WfProcMgr
siebmtshmw 1940     0       13     0.0   0.0   365968   0        eChannelObjMgr_enu
siebmtshmw 2544     0       13     0.0   0.0   365464   0        eChannelObjMgr_enu
siebmtshmw 3208     0       13     0.0   0.0   368188   0        eChannelObjMgr_enu
siebmtshmw 3612     0       13     0.0   0.0   364932   0        eChannelObjMgr_enu

SIEBEL_COMPONENT               CNT   THCNT  %CPU    TOT_MEM   AVG_MEM
================               ====  =====  ====    =======  ========
FINSObjMgr_enu                 18    234     0.0    7108576  394920.9
FSMSrvr                        1     25      0.0      11300   11300.0
SCBroker                       1     1       0.0       8212    8212.0
SRBroker                       1     37      0.0      23372   23372.0
SRProc                         1     8       0.0      96376   96376.0
WfProcMgr                      1     95      0.0     219500  219500.0
eChannelObjMgr_enu             4     52      0.0    1464552  366138.0
                               ====  =====  ====     ======  ========
           Cumulative          27    452       0    8931888
```

x366 System Dual-Core Intel Xeon MP CPUs

Siebel Application Server Configuration

The following is the Korn shell script used to configure the Siebel Application Server in preparation for a 3,600 user run:

```
#!/bin/ksh
# Config Script Version Number: 1

function apply_config {
#
# Global Configuration Variables Used To Determine Parameters.
# Target No. Users: 3700
# No. Siebel App. Servers: 1
# No. Web Servers: 1
# App. Server CPU Bound?: False
# AOM SQL Cursor Cache: 0
# AOM SQL Data Cache: 0
# App. Server OSDF Latches Factor: 4233
# Web Client Threads: 3700
#

#
# Siebel Service Request Broker
#
COMPONENT=SRBroker
echo "change param MaxTasks=100 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MinMTServers=1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MaxMTServers=1 for comp "$COMPONENT >> $COMMAND_FILE

#
# Siebel Server Data Source
#
COMPONENT=ServerDataSrc
echo "change param DSMaxCachedCursors=0 for named subsystem "$COMPONENT >> $COMMAND_FILE
echo "change param DSMaxCachedDataSets=0 for named subsystem "$COMPONENT >> $COMMAND_FILE

#
# AOM Specific Configuration Variables Used To Determine Parameters.
# AOM1 Component Name: FINSObjMgr_enu
# No. Tasks per Thread: 1
# Thread Pooling Enabled?: False
# Thread Affinity Enabled?: True
# No. Users per DB Connection: 1
# Desired Users per AOM: 100
#
COMPONENT=FINSObjMgr_enu
echo "change param MaxTasks=2960 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MinMTServers=30 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MaxMTServers=30 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MinSharedDbConns=-1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MaxSharedDbConns=-1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MinTrxDbConns=-1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MinPoolThreads=0 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MaxPoolThreads=0 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param SessPerSisnConn=20 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param CommEnable=False for comp "$COMPONENT >> $COMMAND_FILE
echo "change param CommConfigManager=True for comp "$COMPONENT >> $COMMAND_FILE
echo "change param UseThreadPool=False for comp "$COMPONENT >> $COMMAND_FILE
echo "change param ThreadAffinity=True for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MemProtection=True for comp "$COMPONENT >> $COMMAND_FILE
echo "change EvtLogLvl EventContext=1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change EvtLogLvl ObjMgrSqlLog=1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change EvtLogLvl SQLFlags=0 for service "$COMPONENT >> $COMMAND_FILE

#
# AOM Specific Configuration Variables Used To Determine Parameters.
# AOM2 Component Name: eChannelObjMgr_enu
# No. Tasks per Thread: 1
# Thread Pooling Enabled?: False
# Thread Affinity Enabled?: True
# No. Users per DB Connection: 1
# Desired Users per AOM: 100
#
COMPONENT=eChannelObjMgr_enu
echo "change param MaxTasks=740 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MinMTServers=7 for comp "$COMPONENT >> $COMMAND_FILE
```

```
echo "change param MaxMTServers=7 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MinSharedDbConns=-1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MaxSharedDbConns=-1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MinTrxDbConns=-1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MinPoolThreads=0 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MaxPoolThreads=0 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param SessPerSisnConn=20 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param CommEnable=False for comp "$COMPONENT >> $COMMAND_FILE
echo "change param CommConfigManager=True for comp "$COMPONENT >> $COMMAND_FILE
echo "change param UseThreadPool=False for comp "$COMPONENT >> $COMMAND_FILE
echo "change param ThreadAffinity=True for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MemProtection=True for comp "$COMPONENT >> $COMMAND_FILE
echo "change EvtLogLvl EventContext=1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change EvtLogLvl ObjMgrSqlLog=1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change EvtLogLvl SQLFlags=0 for service "$COMPONENT >> $COMMAND_FILE

#
# AOM Specific Configuration Variables Used To Determine Parameters.
# AOM3 Component Name: WfProcMgr
# No. Tasks per Thread: 1
# Thread Pooling Enabled?: False
# Thread Affinity Enabled?: True
# No. Users per DB Connection: 2
# Desired Users per AOM: 100
#
COMPONENT=WfProcMgr
echo "change param MaxTasks=148 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MinMTServers=1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MaxMTServers=1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MinSharedDbConns=74 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MaxSharedDbConns=74 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MinTrxDbConns=74 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MinPoolThreads=0 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MaxPoolThreads=0 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param SessPerSisnConn=0 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param HonorMaxTasks=True for comp "$COMPONENT >> $COMMAND_FILE
echo "change param UsePrivatePort=True for comp "$COMPONENT >> $COMMAND_FILE
echo "change param BypassHandler=True for comp "$COMPONENT >> $COMMAND_FILE
echo "change param ModelCacheMax=84 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MemProtection=True for comp "$COMPONENT >> $COMMAND_FILE
echo "change EvtLogLvl EventContext=1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change EvtLogLvl ObjMgrSqlLog=1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change EvtLogLvl SQLFlags=0 for service "$COMPONENT >> $COMMAND_FILE

#
# Apply the configuration changes to the Siebel Application Server
#
"c:/sea77/siebsrvr/bin/srvrmgr" $CONNECT_STRING <<EOF
read $COMMAND_FILE
quit
EOF

printf "\n'srvrmgr' Connect String Used: %s\n" "$CONNECT_STRING"
printf "Application Server Configured: %s\n\n" "$COMPUTERNAME"
}

################################################################################
################################################################################
#
# Main Script Starts Here
#
################################################################################
################################################################################

CONNECT_STRING="-e siebel -g $COMPUTERNAME -u sadmin -p sadmin"
COMMAND_FILE=c:/temp/srvrmgr.$$
OUTPUT_FILE=c:/temp/srvrmgr.out.$$

if [[ -f $COMMAND_FILE ]]; then
   rm $COMMAND_FILE
fi

if [[ -f $OUTPUT_FILE ]]; then
   rm $OUTPUT_FILE
fi

#
# Specify the App Server we want to configure
#
echo "set server "$COMPUTERNAME >> $COMMAND_FILE
```

```
#
# Execute the 'set server' command in 'srvrmgr' and capture the OUTPUT
#
"c:/sea77/siebsrvr/bin/srvrmgr" $CONNECT_STRING <<EOF >$OUTPUT_FILE
read $COMMAND_FILE
quit
EOF

#
# GREP the OUTPUT from the 'srvrmgr' command for our App Server Name in the
# PROMPT. The PROMPT will take the format 'srvrmgr:<App Server Name>', if
# the 'set server' command was successful
#
if [[ -z "`grep \"srvrmgr:$COMPUTERNAME\" $OUTPUT_FILE`" ]]; then
   printf "\n\nERROR! - Siebel App Server '%s' Not Found In Enterprise '%s'!\n\n"
"$COMPUTERNAME"  "siebel"
else
   #
   # App Server exists! Apply the config to it. The FUNCTION continues to APPEND
   # the config details to the COMMAND_FILE, which at the moment, contains just
   # the valid 'set server' command
   #
   apply_config
fi

rm $COMMAND_FILE
rm $OUTPUT_FILE
```

Siebel Run-time Processes

The following is a custom display of the Siebel Application Server processes that were running before the start of the 3,600 user run:

```
COMMAND    PID      SC     THCNT  %CPU  %MEM  VSZ      TASK_ID  SIEBEL_COMPONENT
siebmtshmw 2156     0      7      0.0   0.0   45740    0        FINSObjMgr_enu
siebmtshmw 2180     0      6      0.0   0.0   45724    0        FINSObjMgr_enu
siebmtshmw 2220     0      6      0.0   0.0   45728    0        FINSObjMgr_enu
siebmtshmw 2240     0      6      0.0   0.0   45728    0        FINSObjMgr_enu
siebmtshmw 2468     0      6      0.0   0.0   45728    0        FINSObjMgr_enu
siebmtshmw 2496     0      6      0.0   0.0   45724    0        FINSObjMgr_enu
siebmtshmw 2864     0      6      0.0   0.0   45728    0        FINSObjMgr_enu
siebmtshmw 2940     0      6      0.0   0.0   45724    0        FINSObjMgr_enu
siebmtshmw 3112     0      7      0.0   0.0   45736    0        FINSObjMgr_enu
siebmtshmw 3200     0      7      0.0   0.0   45732    0        FINSObjMgr_enu
siebmtshmw 3216     0      6      0.0   0.0   45724    0        FINSObjMgr_enu
siebmtshmw 3420     0      6      0.0   0.0   45724    0        FINSObjMgr_enu
siebmtshmw 3528     0      6      0.0   0.0   45720    0        FINSObjMgr_enu
siebmtshmw 3544     0      6      0.0   0.0   45732    0        FINSObjMgr_enu
siebmtshmw 3560     0      6      0.0   0.0   45724    0        FINSObjMgr_enu
siebmtshmw 3596     0      6      0.0   0.0   45728    0        FINSObjMgr_enu
siebmtshmw 3600     0      6      0.0   0.0   45724    0        FINSObjMgr_enu
siebmtshmw 3640     0      6      0.0   0.0   45728    0        FINSObjMgr_enu
siebmtshmw 3700     0      6      0.0   0.0   45732    0        FINSObjMgr_enu
siebmtshmw 3724     0      6      0.0   0.0   45732    0        FINSObjMgr_enu
siebmtshmw 3800     0      6      0.0   0.0   45732    0        FINSObjMgr_enu
siebmtshmw 3816     0      6      0.0   0.0   45732    0        FINSObjMgr_enu
siebmtshmw 3884     0      6      0.0   0.0   45728    0        FINSObjMgr_enu
siebmtshmw 3908     0      7      0.0   0.0   45740    0        FINSObjMgr_enu
siebmtshmw 3960     0      6      0.0   0.0   45724    0        FINSObjMgr_enu
siebmtshmw 3996     0      6      0.0   0.0   45736    0        FINSObjMgr_enu
siebmtshmw 4028     0      6      0.0   0.0   45732    0        FINSObjMgr_enu
siebmtshmw 4072     0      6      0.0   0.0   45732    0        FINSObjMgr_enu
siebmtshmw 416      0      6      0.0   0.0   45724    0        FINSObjMgr_enu
siebmtshmw 4948     0      6      0.0   0.0   45720    0        FINSObjMgr_enu
siebmtsh   2088     0      25     0.0   0.0   11224    0        FSMSrvr
siebproc   2112     0      1      0.0   0.0   8252     0        SCBroker
siebmtsh   2116     0      17     0.0   0.0   20996    0        SRBroker
siebmtsh   2196     0      8      0.0   0.0   96740    0        SRProc
siebmtshmw 2476     0      154    0.0   0.0   164680   0        WfProcMgr
siebmtshmw 1576     0      6      0.0   0.0   26564    0        eChannelObjMgr_enu
siebmtshmw 2856     0      6      0.0   0.0   26556    0        eChannelObjMgr_enu
siebmtshmw 3120     0      6      0.0   0.0   26556    0        eChannelObjMgr_enu
siebmtshmw 3272     0      6      0.0   0.0   26556    0        eChannelObjMgr_enu
siebmtshmw 4172     0      6      0.0   0.0   26552    0        eChannelObjMgr_enu
siebmtshmw 4972     0      6      0.0   0.0   26552    0        eChannelObjMgr_enu
siebmtshmw 5024     0      6      0.0   0.0   26552    0        eChannelObjMgr_enu

SIEBEL_COMPONENT              CNT   THCNT  %CPU    TOT_MEM   AVG_MEM
================              ====  =====  ====    =======  ========
FINSObjMgr_enu                30    184     0.0    1371860   45728.7
FSMSrvr                       1     25      0.0      11224   11224.0
SCBroker                      1     1       0.0       8252    8252.0
SRBroker                      1     17      0.0      20996   20996.0
SRProc                        1     8       0.0      96740   96740.0
WfProcMgr                     1     154     0.0     164680  164680.0
eChannelObjMgr_enu            7     42      0.0     185888   26555.4
                              ====  =====  ====     ======  ========
           Cumulative         42    431       0    1859640
```

The following are the Siebel Application Server processes that were running after the 3,600 user run:

```
COMMAND    PID      SC      THCNT  %CPU  %MEM  VSZ      TASK_ID  SIEBEL_COMPONENT
siebmtshmw 2156     0       13     0.0   0.0   414268   0        FINSObjMgr_enu
siebmtshmw 2180     0       13     0.0   0.0   414548   0        FINSObjMgr_enu
siebmtshmw 2220     0       13     0.0   0.0   418488   0        FINSObjMgr_enu
siebmtshmw 2240     0       13     0.0   0.0   413848   0        FINSObjMgr_enu
siebmtshmw 2468     0       13     0.0   0.0   412556   0        FINSObjMgr_enu
siebmtshmw 2496     0       13     0.0   0.0   415324   0        FINSObjMgr_enu
siebmtshmw 2864     0       13     0.0   0.0   415392   0        FINSObjMgr_enu
siebmtshmw 2940     0       13     0.0   0.0   414324   0        FINSObjMgr_enu
siebmtshmw 3112     0       13     0.0   0.0   413992   0        FINSObjMgr_enu
siebmtshmw 3200     0       13     0.0   0.0   414948   0        FINSObjMgr_enu
siebmtshmw 3216     0       13     0.0   0.0   415264   0        FINSObjMgr_enu
siebmtshmw 3420     0       13     0.0   0.0   415728   0        FINSObjMgr_enu
siebmtshmw 3528     0       13     0.0   0.0   410568   0        FINSObjMgr_enu
siebmtshmw 3544     0       13     0.0   0.0   414060   0        FINSObjMgr_enu
siebmtshmw 3560     0       13     0.0   0.0   413020   0        FINSObjMgr_enu
siebmtshmw 3596     0       13     0.0   0.0   413604   0        FINSObjMgr_enu
siebmtshmw 3600     0       13     0.0   0.0   416444   0        FINSObjMgr_enu
siebmtshmw 3640     0       13     0.0   0.0   415016   0        FINSObjMgr_enu
siebmtshmw 3700     0       13     0.0   0.0   419244   0        FINSObjMgr_enu
siebmtshmw 3724     0       13     0.0   0.0   413272   0        FINSObjMgr_enu
siebmtshmw 3800     0       13     0.0   0.0   413972   0        FINSObjMgr_enu
siebmtshmw 3816     0       13     0.0   0.0   412716   0        FINSObjMgr_enu
siebmtshmw 3884     0       13     0.0   0.0   415212   0        FINSObjMgr_enu
siebmtshmw 3908     0       13     0.0   0.0   418468   0        FINSObjMgr_enu
siebmtshmw 3960     0       13     0.0   0.0   413216   0        FINSObjMgr_enu
siebmtshmw 3996     0       13     0.0   0.0   415216   0        FINSObjMgr_enu
siebmtshmw 4028     0       13     0.0   0.0   413404   0        FINSObjMgr_enu
siebmtshmw 4072     0       13     0.0   0.0   414508   0        FINSObjMgr_enu
siebmtshmw 416      0       13     0.0   0.0   413416   0        FINSObjMgr_enu
siebmtshmw 4948     0       13     0.0   0.0   412596   0        FINSObjMgr_enu
siebmtsh   2088     0       25     0.0   0.0   11224    0        FSMSrvr
siebproc   2112     0       1      0.0   0.0   8620     0        SCBroker
siebmtsh   2116     0       52     0.0   0.0   26156    0        SRBroker
siebmtsh   2196     0       8      0.0   0.0   96788    0        SRProc
siebmtshmw 2476     0       155    0.0   0.0   260556   0        WfProcMgr
siebmtshmw 1576     0       13     0.0   0.0   372172   0        eChannelObjMgr_enu
siebmtshmw 2856     0       13     0.0   0.0   369900   0        eChannelObjMgr_enu
siebmtshmw 3120     0       13     0.0   0.0   368376   0        eChannelObjMgr_enu
siebmtshmw 3272     0       13     0.0   0.0   368272   0        eChannelObjMgr_enu
siebmtshmw 4172     0       13     0.0   0.0   368652   0        eChannelObjMgr_enu
siebmtshmw 4972     0       13     0.0   0.0   372656   0        eChannelObjMgr_enu
siebmtshmw 5024     0       13     0.0   0.0   366416   0        eChannelObjMgr_enu

SIEBEL_COMPONENT               CNT   THCNT  %CPU    TOT_MEM   AVG_MEM
================               ====  =====  ====    =======  ========
FINSObjMgr_enu                 30    390     0.0   12436632  414554.4
FSMSrvr                        1     25      0.0      11224   11224.0
SCBroker                       1     1       0.0       8620    8620.0
SRBroker                       1     52      0.0      26156   26156.0
SRProc                         1     8       0.0      96788   96788.0
WfProcMgr                      1     155     0.0     260556  260556.0
eChannelObjMgr_enu             7     91      0.0    2586444  369492.0
                               ====  =====  ====     ======  ========
           Cumulative          42    722       0   15426420
```

x460 System Single-Core Intel Xeon MP CPUs (One Chassis)

Siebel Application Server Configuration

The following is the Korn shell script used to configure the Siebel Application Server in preparation for a 2,000 user run:

```
#!/bin/ksh
# Config Script Version Number: 1

function apply_config {
#
# Global Configuration Variables Used To Determine Parameters.
# Target No. Users: 2100
# No. Siebel App. Servers: 1
# No. Web Servers: 1
# App. Server CPU Bound?: False
# AOM SQL Cursor Cache: 0
# AOM SQL Data Cache: 0
# App. Server OSDF Latches Factor: 2402
# Web Client Threads: 2100
#

#
# Siebel Service Request Broker
#
COMPONENT=SRBroker
echo "change param MaxTasks=100 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MinMTServers=1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MaxMTServers=1 for comp "$COMPONENT >> $COMMAND_FILE

#
# Siebel Server Data Source
#
COMPONENT=ServerDataSrc
echo "change param DSMaxCachedCursors=0 for named subsystem "$COMPONENT >> $COMMAND_FILE
echo "change param DSMaxCachedDataSets=0 for named subsystem "$COMPONENT >> $COMMAND_FILE

#
# AOM Specific Configuration Variables Used To Determine Parameters.
# AOM1 Component Name: FINSObjMgr_enu
# No. Tasks per Thread: 1
# Thread Pooling Enabled?: False
# Thread Affinity Enabled?: True
# No. Users per DB Connection: 1
# Desired Users per AOM: 100
#
COMPONENT=FINSObjMgr_enu
echo "change param MaxTasks=1680 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MinMTServers=17 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MaxMTServers=17 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MinSharedDbConns=-1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MaxSharedDbConns=-1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MinTrxDbConns=-1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MinPoolThreads=0 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MaxPoolThreads=0 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param SessPerSisnConn=20 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param CommEnable=False for comp "$COMPONENT >> $COMMAND_FILE
echo "change param CommConfigManager=True for comp "$COMPONENT >> $COMMAND_FILE
echo "change param UseThreadPool=False for comp "$COMPONENT >> $COMMAND_FILE
echo "change param ThreadAffinity=True for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MemProtection=True for comp "$COMPONENT >> $COMMAND_FILE
echo "change EvtLogLvl EventContext=1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change EvtLogLvl ObjMgrSqlLog=1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change EvtLogLvl SQLFlags=0 for service "$COMPONENT >> $COMMAND_FILE

#
# AOM Specific Configuration Variables Used To Determine Parameters.
# AOM2 Component Name: eChannelObjMgr_enu
# No. Tasks per Thread: 1
# Thread Pooling Enabled?: False
# Thread Affinity Enabled?: True
# No. Users per DB Connection: 1
# Desired Users per AOM: 100
#
COMPONENT=eChannelObjMgr_enu
echo "change param MaxTasks=420 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MinMTServers=4 for comp "$COMPONENT >> $COMMAND_FILE
```

```
echo "change param MaxMTServers=4 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MinSharedDbConns=-1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MaxSharedDbConns=-1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MinTrxDbConns=-1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MinPoolThreads=0 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MaxPoolThreads=0 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param SessPerSisnConn=20 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param CommEnable=False for comp "$COMPONENT >> $COMMAND_FILE
echo "change param CommConfigManager=True for comp "$COMPONENT >> $COMMAND_FILE
echo "change param UseThreadPool=False for comp "$COMPONENT >> $COMMAND_FILE
echo "change param ThreadAffinity=True for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MemProtection=True for comp "$COMPONENT >> $COMMAND_FILE
echo "change EvtLogLvl EventContext=1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change EvtLogLvl ObjMgrSqlLog=1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change EvtLogLvl SQLFlags=0 for service "$COMPONENT >> $COMMAND_FILE

#
# AOM Specific Configuration Variables Used To Determine Parameters.
# AOM3 Component Name: WfProcMgr
# No. Tasks per Thread: 1
# Thread Pooling Enabled?: False
# Thread Affinity Enabled?: True
# No. Users per DB Connection: 2
# Desired Users per AOM: 100
#
COMPONENT=WfProcMgr
echo "change param MaxTasks=84 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MinMTServers=1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MaxMTServers=1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MinSharedDbConns=42 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MaxSharedDbConns=42 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MinTrxDbConns=42 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MinPoolThreads=0 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MaxPoolThreads=0 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param SessPerSisnConn=0 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param HonorMaxTasks=True for comp "$COMPONENT >> $COMMAND_FILE
echo "change param UsePrivatePort=True for comp "$COMPONENT >> $COMMAND_FILE
echo "change param BypassHandler=True for comp "$COMPONENT >> $COMMAND_FILE
echo "change param ModelCacheMax=84 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MemProtection=True for comp "$COMPONENT >> $COMMAND_FILE
echo "change EvtLogLvl EventContext=1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change EvtLogLvl ObjMgrSqlLog=1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change EvtLogLvl SQLFlags=0 for service "$COMPONENT >> $COMMAND_FILE

#
# Apply the configuration changes to the Siebel Application Server
#
"c:/sea77/siebsrvr/bin/srvrmgr" $CONNECT_STRING <<EOF
read $COMMAND_FILE
quit
EOF

printf "\n'srvrmgr' Connect String Used: %s\n" "$CONNECT_STRING"
printf "Application Server Configured: %s\n\n" "$COMPUTERNAME"
}

################################################################################
################################################################################
#
# Main Script Starts Here
#
################################################################################
################################################################################

CONNECT_STRING="-e siebel -g $COMPUTERNAME -u sadmin -p sadmin"
COMMAND_FILE=c:/temp/srvrmgr.$$
OUTPUT_FILE=c:/temp/srvrmgr.out.$$

if [[ -f $COMMAND_FILE ]]; then
   rm $COMMAND_FILE
fi

if [[ -f $OUTPUT_FILE ]]; then
   rm $OUTPUT_FILE
fi

#
# Specify the App Server we want to configure
#
echo "set server "$COMPUTERNAME >> $COMMAND_FILE
```

```
#
# Execute the 'set server' command in 'srvrmgr' and capture the OUTPUT
#
"c:/sea77/siebsrvr/bin/srvrmgr" $CONNECT_STRING <<EOF >$OUTPUT_FILE
read $COMMAND_FILE
quit
EOF

#
# GREP the OUTPUT from the 'srvrmgr' command for our App Server Name in the
# PROMPT. The PROMPT will take the format 'srvrmgr:<App Server Name>', if
# the 'set server' command was successful
#
if [[ -z "`grep \"srvrmgr:$COMPUTERNAME\" $OUTPUT_FILE`" ]]; then
   printf "\n\nERROR! - Siebel App Server '%s' Not Found In Enterprise '%s'!\n\n"
"$COMPUTERNAME"  "siebel"
else
   #
   # App Server exists! Apply the config to it. The FUNCTION continues to APPEND
   # the config details to the COMMAND_FILE, which at the moment, contains just
   # the valid 'set server' command
   #
   apply_config
fi

rm $COMMAND_FILE
rm $OUTPUT_FILE
```

Siebel Run-time Processes

The following is a custom display of the Siebel Application Server processes that were running before the start of the 2,000 user run:

```
COMMAND    PID      SC     THCNT  %CPU  %MEM  VSZ      TASK_ID  SIEBEL_COMPONENT
siebmtshmw 11564    0      6      0.0   0.0   34308    0        FINSObjMgr_enu
siebmtshmw 12588    0      6      0.0   0.0   34308    0        FINSObjMgr_enu
siebmtshmw 14080    0      6      0.0   0.0   34308    0        FINSObjMgr_enu
siebmtshmw 14096    0      6      0.0   0.0   34308    0        FINSObjMgr_enu
siebmtshmw 1836     0      6      0.0   0.0   34312    0        FINSObjMgr_enu
siebmtshmw 2060     0      7      0.0   0.0   34308    0        FINSObjMgr_enu
siebmtshmw 2168     0      6      0.0   0.0   34308    0        FINSObjMgr_enu
siebmtshmw 2344     0      7      0.0   0.0   34316    0        FINSObjMgr_enu
siebmtshmw 2516     0      7      0.0   0.0   34304    0        FINSObjMgr_enu
siebmtshmw 3932     0      6      0.0   0.0   34304    0        FINSObjMgr_enu
siebmtshmw 3964     0      6      0.0   0.0   34308    0        FINSObjMgr_enu
siebmtshmw 4316     0      6      0.0   0.0   34304    0        FINSObjMgr_enu
siebmtshmw 5912     0      6      0.0   0.0   34312    0        FINSObjMgr_enu
siebmtshmw 6352     0      6      0.0   0.0   34300    0        FINSObjMgr_enu
siebmtshmw 828      0      6      0.0   0.0   34304    0        FINSObjMgr_enu
siebmtshmw 852      0      7      0.0   0.0   34312    0        FINSObjMgr_enu
siebmtshmw 920      0      6      0.0   0.0   34304    0        FINSObjMgr_enu
siebmtsh   2636     0      25     0.0   0.0   10864    0        FSMSrvr
siebproc   3200     0      1      0.0   0.0   7884     0        SCBroker
siebmtsh   9696     0      17     0.0   0.0   20580    0        SRBroker
siebmtsh   2004     0      8      0.0   0.0   96332    0        SRProc
siebmtshmw 2428     0      90     0.0   0.0   154372   0        WfProcMgr
siebmtshmw 2164     0      6      0.0   0.0   23432    0        eChannelObjMgr_enu
siebmtshmw 2712     0      6      0.0   0.0   23416    0        eChannelObjMgr_enu
siebmtshmw 3004     0      6      0.0   0.0   23432    0        eChannelObjMgr_enu
siebmtshmw 5960     0      6      0.0   0.0   23436    0        eChannelObjMgr_enu

SIEBEL_COMPONENT              CNT   THCNT  %CPU    TOT_MEM   AVG_MEM
================              ====  =====  ====    =======  ========
FINSObjMgr_enu                17    106     0.0     583228   34307.5
FSMSrvr                       1     25      0.0      10864   10864.0
SCBroker                      1     1       0.0       7884    7884.0
SRBroker                      1     17      0.0      20580   20580.0
SRProc                        1     8       0.0      96332   96332.0
WfProcMgr                     1     90      0.0     154372  154372.0
eChannelObjMgr_enu            4     24      0.0      93716   23429.0
                              ====  =====  ====     ======  ========
           Cumulative         26    271       0     966976
```

The following are the Siebel Application Server processes that were running after the 2,000 user run:

```
COMMAND    PID      SC     THCNT  %CPU  %MEM  VSZ      TASK_ID  SIEBEL_COMPONENT
siebmtshmw 11564    0      13     0.0   0.0   394900   0        FINSObjMgr_enu
siebmtshmw 12588    0      13     0.0   0.0   396040   0        FINSObjMgr_enu
siebmtshmw 14080    0      13     0.0   0.0   390672   0        FINSObjMgr_enu
siebmtshmw 14096    0      13     0.0   0.0   395828   0        FINSObjMgr_enu
siebmtshmw 1836     0      13     0.0   0.0   394788   0        FINSObjMgr_enu
siebmtshmw 2060     0      13     0.0   0.0   394240   0        FINSObjMgr_enu
siebmtshmw 2168     0      13     0.0   0.0   398712   0        FINSObjMgr_enu
siebmtshmw 2344     0      13     0.0   0.0   396304   0        FINSObjMgr_enu
siebmtshmw 2516     0      13     0.0   0.0   396144   0        FINSObjMgr_enu
siebmtshmw 3932     0      13     0.0   0.0   393240   0        FINSObjMgr_enu
siebmtshmw 3964     0      13     0.0   0.0   396396   0        FINSObjMgr_enu
siebmtshmw 4316     0      13     0.0   0.0   396556   0        FINSObjMgr_enu
siebmtshmw 5912     0      13     0.0   0.0   396552   0        FINSObjMgr_enu
siebmtshmw 6352     0      13     0.0   0.0   395408   0        FINSObjMgr_enu
siebmtshmw 828      0      13     0.0   0.0   395312   0        FINSObjMgr_enu
siebmtshmw 852      0      13     0.0   0.0   394236   0        FINSObjMgr_enu
siebmtshmw 920      0      13     0.0   0.0   395892   0        FINSObjMgr_enu
siebmtsh   2636     0      25     0.0   0.0   10864    0        FSMSrvr
siebproc   3200     0      1      0.0   0.0   8176     0        SCBroker
siebmtsh   9696     0      36     0.0   0.0   23200    0        SRBroker
siebmtsh   2004     0      8      0.0   0.0   96356    0        SRProc
siebmtshmw 2428     0      91     0.0   0.0   208352   0        WfProcMgr
siebmtshmw 2164     0      13     0.0   0.0   358016   0        eChannelObjMgr_enu
siebmtshmw 2712     0      13     0.0   0.0   358360   0        eChannelObjMgr_enu
siebmtshmw 3004     0      13     0.0   0.0   361004   0        eChannelObjMgr_enu
siebmtshmw 5960     0      13     0.0   0.0   362032   0        eChannelObjMgr_enu

SIEBEL_COMPONENT              CNT    THCNT  %CPU    TOT_MEM   AVG_MEM
================              ====   =====  ====    =======   ========
FINSObjMgr_enu                17     221     0.0    6721220  395365.9
FSMSrvr                       1      25      0.0      10864   10864.0
SCBroker                      1      1       0.0       8176    8176.0
SRBroker                      1      36      0.0      23200   23200.0
SRProc                        1      8       0.0      96356   96356.0
WfProcMgr                     1      91      0.0     208352  208352.0
eChannelObjMgr_enu            4      52      0.0    1439412  359853.0
                              ====   =====  ====     ======   ========
           Cumulative         26     434       0    8507580
```

x460 System Single-Core Intel Xeon MP CPUs (Two Chassis, Single Partition)

Siebel Application Server Configuration

The following is the Korn shell script used to configure the Siebel Application Server in preparation for a 3,400 user run:

```
#!/bin/ksh
# Config Script Version Number: 1

function apply_config {
#
# Global Configuration Variables Used To Determine Parameters.
# Target No. Users: 3500
# No. Siebel App. Servers: 1
# No. Web Servers: 1
# App. Server CPU Bound?: False
# AOM SQL Cursor Cache: 0
# AOM SQL Data Cache: 0
# App. Server OSDF Latches Factor: 4004
# Web Client Threads: 3500
#

#
# Siebel Service Request Broker
#
COMPONENT=SRBroker
echo "change param MaxTasks=100 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MinMTServers=1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MaxMTServers=1 for comp "$COMPONENT >> $COMMAND_FILE

#
# Siebel Server Data Source
#
COMPONENT=ServerDataSrc
echo "change param DSMaxCachedCursors=0 for named subsystem "$COMPONENT >> $COMMAND_FILE
echo "change param DSMaxCachedDataSets=0 for named subsystem "$COMPONENT >> $COMMAND_FILE

#
# AOM Specific Configuration Variables Used To Determine Parameters.
# AOM1 Component Name: FINSObjMgr_enu
# No. Tasks per Thread: 1
# Thread Pooling Enabled?: False
# Thread Affinity Enabled?: True
# No. Users per DB Connection: 1
# Desired Users per AOM: 100
#
COMPONENT=FINSObjMgr_enu
echo "change param MaxTasks=2800 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MinMTServers=28 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MaxMTServers=28 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MinSharedDbConns=-1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MaxSharedDbConns=-1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MinTrxDbConns=-1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MinPoolThreads=0 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MaxPoolThreads=0 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param SessPerSisnConn=20 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param CommEnable=False for comp "$COMPONENT >> $COMMAND_FILE
echo "change param CommConfigManager=True for comp "$COMPONENT >> $COMMAND_FILE
echo "change param UseThreadPool=False for comp "$COMPONENT >> $COMMAND_FILE
echo "change param ThreadAffinity=True for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MemProtection=True for comp "$COMPONENT >> $COMMAND_FILE
echo "change EvtLogLvl EventContext=1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change EvtLogLvl ObjMgrSqlLog=1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change EvtLogLvl SQLFlags=0 for service "$COMPONENT >> $COMMAND_FILE

#
# AOM Specific Configuration Variables Used To Determine Parameters.
# AOM2 Component Name: eChannelObjMgr_enu
# No. Tasks per Thread: 1
# Thread Pooling Enabled?: False
# Thread Affinity Enabled?: True
# No. Users per DB Connection: 1
# Desired Users per AOM: 100
#
COMPONENT=eChannelObjMgr_enu
echo "change param MaxTasks=700 for comp "$COMPONENT >> $COMMAND_FILE
```

```
echo "change param MinMTServers=7 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MaxMTServers=7 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MinSharedDbConns=-1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MaxSharedDbConns=-1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MinTrxDbConns=-1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MinPoolThreads=0 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MaxPoolThreads=0 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param SessPerSisnConn=20 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param CommEnable=False for comp "$COMPONENT >> $COMMAND_FILE
echo "change param CommConfigManager=True for comp "$COMPONENT >> $COMMAND_FILE
echo "change param UseThreadPool=False for comp "$COMPONENT >> $COMMAND_FILE
echo "change param ThreadAffinity=True for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MemProtection=True for comp "$COMPONENT >> $COMMAND_FILE
echo "change EvtLogLvl EventContext=1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change EvtLogLvl ObjMgrSqlLog=1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change EvtLogLvl SQLFlags=0 for service "$COMPONENT >> $COMMAND_FILE

#
# AOM Specific Configuration Variables Used To Determine Parameters.
# AOM3 Component Name: WfProcMgr
# No. Tasks per Thread: 1
# Thread Pooling Enabled?: False
# Thread Affinity Enabled?: True
# No. Users per DB Connection: 2
# Desired Users per AOM: 100
#
COMPONENT=WfProcMgr
echo "change param MaxTasks=140 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MinMTServers=1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MaxMTServers=1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MinSharedDbConns=70 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MaxSharedDbConns=70 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MinTrxDbConns=70 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MinPoolThreads=0 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MaxPoolThreads=0 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param SessPerSisnConn=0 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param HonorMaxTasks=True for comp "$COMPONENT >> $COMMAND_FILE
echo "change param UsePrivatePort=True for comp "$COMPONENT >> $COMMAND_FILE
echo "change param BypassHandler=True for comp "$COMPONENT >> $COMMAND_FILE
echo "change param ModelCacheMax=84 for comp "$COMPONENT >> $COMMAND_FILE
echo "change param MemProtection=True for comp "$COMPONENT >> $COMMAND_FILE
echo "change EvtLogLvl EventContext=1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change EvtLogLvl ObjMgrSqlLog=1 for comp "$COMPONENT >> $COMMAND_FILE
echo "change EvtLogLvl SQLFlags=0 for service "$COMPONENT >> $COMMAND_FILE

#
# Apply the configuration changes to the Siebel Application Server
#
"c:/sea77/siebsrvr/bin/srvrmgr" $CONNECT_STRING <<EOF
read $COMMAND_FILE
quit
EOF

printf "\n'srvrmgr' Connect String Used: %s\n" "$CONNECT_STRING"
printf "Application Server Configured: %s\n\n" "$COMPUTERNAME"
}

################################################################################
################################################################################
#
# Main Script Starts Here
#
################################################################################
################################################################################

CONNECT_STRING="-e siebel -g $COMPUTERNAME -u sadmin -p sadmin"
COMMAND_FILE=c:/temp/srvrmgr.$$
OUTPUT_FILE=c:/temp/srvrmgr.out.$$

if [[ -f $COMMAND_FILE ]]; then
   rm $COMMAND_FILE
fi

if [[ -f $OUTPUT_FILE ]]; then
   rm $OUTPUT_FILE
fi

#
# Specify the App Server we want to configure
#
```

```
echo "set server "$COMPUTERNAME >> $COMMAND_FILE

#
# Execute the 'set server' command in 'srvrmgr' and capture the OUTPUT
#
"c:/sea77/siebsrvr/bin/srvrmgr" $CONNECT_STRING <<EOF >$OUTPUT_FILE
read $COMMAND_FILE
quit
EOF

#
# GREP the OUTPUT from the 'srvrmgr' command for our App Server Name in the
# PROMPT. The PROMPT will take the format 'srvrmgr:<App Server Name>', if
# the 'set server' command was successful
#
if [[ -z "`grep \"srvrmgr:$COMPUTERNAME\" $OUTPUT_FILE`" ]]; then
   printf "\n\nERROR! - Siebel App Server '%s' Not Found In Enterprise '%s'!\n\n"
"$COMPUTERNAME"  "siebel"
else
   #
   # App Server exists! Apply the config to it. The FUNCTION continues to APPEND
   # the config details to the COMMAND_FILE, which at the moment, contains just
   # the valid 'set server' command
   #
   apply_config
fi

rm $COMMAND_FILE
rm $OUTPUT_FILE
```

Siebel Run-time Processes

The following is a custom display of the Siebel Application Server processes that were running before the start of the 3,400 user run:

```
COMMAND    PID      SC     THCNT  %CPU  %MEM  VSZ      TASK_ID  SIEBEL_COMPONENT
siebmtshmw 10116    0      6      0.0   0.0   44292    0        FINSObjMgr_enu
siebmtshmw 10256    0      6      0.0   0.0   44292    0        FINSObjMgr_enu
siebmtshmw 10760    0      6      0.0   0.0   44296    0        FINSObjMgr_enu
siebmtshmw 10848    0      6      0.0   0.0   44292    0        FINSObjMgr_enu
siebmtshmw 11428    0      6      0.0   0.0   44292    0        FINSObjMgr_enu
siebmtshmw 12332    0      6      0.0   0.0   44292    0        FINSObjMgr_enu
siebmtshmw 13052    0      6      0.0   0.0   44292    0        FINSObjMgr_enu
siebmtshmw 13120    0      6      0.0   0.0   44292    0        FINSObjMgr_enu
siebmtshmw 13900    0      7      0.0   0.0   44304    0        FINSObjMgr_enu
siebmtshmw 14504    0      7      0.0   0.0   44304    0        FINSObjMgr_enu
siebmtshmw 14608    0      6      0.0   0.0   44292    0        FINSObjMgr_enu
siebmtshmw 17336    0      6      0.0   0.0   44288    0        FINSObjMgr_enu
siebmtshmw 17584    0      6      0.0   0.0   44292    0        FINSObjMgr_enu
siebmtshmw 18236    0      6      0.0   0.0   44296    0        FINSObjMgr_enu
siebmtshmw 20104    0      6      0.0   0.0   44300    0        FINSObjMgr_enu
siebmtshmw 20464    0      6      0.0   0.0   44292    0        FINSObjMgr_enu
siebmtshmw 2180     0      6      0.0   0.0   44296    0        FINSObjMgr_enu
siebmtshmw 3424     0      6      0.0   0.0   44292    0        FINSObjMgr_enu
siebmtshmw 3836     0      6      0.0   0.0   44300    0        FINSObjMgr_enu
siebmtshmw 3996     0      6      0.0   0.0   44292    0        FINSObjMgr_enu
siebmtshmw 4340     0      6      0.0   0.0   44288    0        FINSObjMgr_enu
siebmtshmw 4436     0      7      0.0   0.0   44312    0        FINSObjMgr_enu
siebmtshmw 5468     0      6      0.0   0.0   44292    0        FINSObjMgr_enu
siebmtshmw 6096     0      6      0.0   0.0   44288    0        FINSObjMgr_enu
siebmtshmw 7160     0      7      0.0   0.0   44304    0        FINSObjMgr_enu
siebmtshmw 7192     0      6      0.0   0.0   44292    0        FINSObjMgr_enu
siebmtshmw 7508     0      6      0.0   0.0   44292    0        FINSObjMgr_enu
siebmtshmw 8256     0      6      0.0   0.0   44292    0        FINSObjMgr_enu
siebmtsh   5248     0      25     0.0   0.0   11180    0        FSMSrvr
siebproc   5964     0      1      0.0   0.0   8220     0        SCBroker
siebmtsh   16028    0      17     0.0   0.0   20956    0        SRBroker
siebmtsh   17324    0      8      0.0   0.0   96712    0        SRProc
siebmtshmw 12012    0      146    0.0   0.0   164020   0        WfProcMgr
siebmtshmw 11120    0      6      0.0   0.0   26184    0        eChannelObjMgr_enu
siebmtshmw 12456    0      6      0.0   0.0   26188    0        eChannelObjMgr_enu
siebmtshmw 13472    0      6      0.0   0.0   26184    0        eChannelObjMgr_enu
siebmtshmw 16308    0      6      0.0   0.0   26184    0        eChannelObjMgr_enu
siebmtshmw 18548    0      6      0.0   0.0   26184    0        eChannelObjMgr_enu
siebmtshmw 19468    0      6      0.0   0.0   26180    0        eChannelObjMgr_enu
siebmtshmw 5916     0      6      0.0   0.0   26184    0        eChannelObjMgr_enu

SIEBEL_COMPONENT              CNT    THCNT  %CPU    TOT_MEM   AVG_MEM
================              ====   =====  ====    =======  ========
FINSObjMgr_enu                28     172     0.0    1240248   44294.6
FSMSrvr                       1      25      0.0      11180   11180.0
SCBroker                      1      1       0.0       8220    8220.0
SRBroker                      1      17      0.0      20956   20956.0
SRProc                        1      8       0.0      96712   96712.0
WfProcMgr                     1      146     0.0     164020  164020.0
eChannelObjMgr_enu            7      42      0.0     183288   26184.0
                              ====   =====  ====     ======  ========
           Cumulative         40     411       0    1724624
```

The following are the Siebel Application Server processes that were running after the 3,400 user run:

```
COMMAND    PID      SC     THCNT  %CPU  %MEM  VSZ      TASK_ID  SIEBEL_COMPONENT
siebmtshmw 10116    0      12     0.0   0.0   414348   0        FINSObjMgr_enu
siebmtshmw 10256    0      12     0.0   0.0   418400   0        FINSObjMgr_enu
siebmtshmw 10760    0      12     0.0   0.0   416076   0        FINSObjMgr_enu
siebmtshmw 10848    0      12     0.0   0.0   415904   0        FINSObjMgr_enu
siebmtshmw 11428    0      12     0.0   0.0   414640   0        FINSObjMgr_enu
siebmtshmw 12332    0      12     0.0   0.0   414872   0        FINSObjMgr_enu
siebmtshmw 13052    0      12     0.0   0.0   415000   0        FINSObjMgr_enu
siebmtshmw 13120    0      12     0.0   0.0   412940   0        FINSObjMgr_enu
siebmtshmw 13900    0      12     0.0   0.0   417608   0        FINSObjMgr_enu
siebmtshmw 14504    0      12     0.0   0.0   419200   0        FINSObjMgr_enu
siebmtshmw 14608    0      12     0.0   0.0   414296   0        FINSObjMgr_enu
siebmtshmw 17336    0      12     0.0   0.0   414980   0        FINSObjMgr_enu
siebmtshmw 17584    0      12     0.0   0.0   419068   0        FINSObjMgr_enu
siebmtshmw 18236    0      12     0.0   0.0   414364   0        FINSObjMgr_enu
siebmtshmw 20104    0      12     0.0   0.0   413340   0        FINSObjMgr_enu
siebmtshmw 20464    0      12     0.0   0.0   414364   0        FINSObjMgr_enu
siebmtshmw 2180     0      12     0.0   0.0   415924   0        FINSObjMgr_enu
siebmtshmw 3424     0      12     0.0   0.0   415476   0        FINSObjMgr_enu
siebmtshmw 3836     0      12     0.0   0.0   416828   0        FINSObjMgr_enu
siebmtshmw 3996     0      12     0.0   0.0   416780   0        FINSObjMgr_enu
siebmtshmw 4340     0      12     0.0   0.0   414212   0        FINSObjMgr_enu
siebmtshmw 4436     0      12     0.0   0.0   416512   0        FINSObjMgr_enu
siebmtshmw 5468     0      12     0.0   0.0   412228   0        FINSObjMgr_enu
siebmtshmw 6096     0      12     0.0   0.0   417628   0        FINSObjMgr_enu
siebmtshmw 7160     0      12     0.0   0.0   416144   0        FINSObjMgr_enu
siebmtshmw 7192     0      12     0.0   0.0   415288   0        FINSObjMgr_enu
siebmtshmw 7508     0      12     0.0   0.0   412524   0        FINSObjMgr_enu
siebmtshmw 8256     0      12     0.0   0.0   415616   0        FINSObjMgr_enu
siebmtsh   5248     0      25     0.0   0.0   11180    0        FSMSrvr
siebproc   5964     0      1      0.0   0.0   8568     0        SCBroker
siebmtsh   16028    0      50     0.0   0.0   25788    0        SRBroker
siebmtsh   17324    0      8      0.0   0.0   96760    0        SRProc
siebmtshmw 12012    0      147    0.0   0.0   264368   0        WfProcMgr
siebmtshmw 11120    0      12     0.0   0.0   361600   0        eChannelObjMgr_enu
siebmtshmw 12456    0      12     0.0   0.0   361972   0        eChannelObjMgr_enu
siebmtshmw 13472    0      12     0.0   0.0   362776   0        eChannelObjMgr_enu
siebmtshmw 16308    0      12     0.0   0.0   361540   0        eChannelObjMgr_enu
siebmtshmw 18548    0      12     0.0   0.0   360988   0        eChannelObjMgr_enu
siebmtshmw 19468    0      12     0.0   0.0   360468   0        eChannelObjMgr_enu
siebmtshmw 5916     0      12     0.0   0.0   360244   0        eChannelObjMgr_enu

SIEBEL_COMPONENT               CNT    THCNT  %CPU    TOT_MEM   AVG_MEM
================               ====   =====  ====    =======   ========
FINSObjMgr_enu                 28     336     0.0   11634560  415520.0
FSMSrvr                        1      25      0.0      11180   11180.0
SCBroker                       1      1       0.0       8568    8568.0
SRBroker                       1      50      0.0      25788   25788.0
SRProc                         1      8       0.0      96760   96760.0
WfProcMgr                      1      147     0.0     264368  264368.0
eChannelObjMgr_enu             7      84      0.0    2529588  361369.7
                               ====   =====  ====     ======   ========
           Cumulative          40     651       0   14570812
```

Appendix H: Database Configuration Parameters

The following sections show the database configuration settings that were in effect for all the Intel performance and PSPP benchmark runs.

Database Configuration

```
 Database configuration release level                  = 0x0a00
 Database release level                                = 0x0a00

 Database territory                                    = US
 Database code page                                    = 1208
 Database code set                                     = UTF-8
 Database country/region code                          = 1
 Database collating sequence                           = BINARY
 Alternate collating sequence              (ALT_COLLATE) =

 Dynamic SQL Query management           (DYN_QUERY_MGMT) = DISABLE

 Discovery support for this database       (DISCOVER_DB) = ENABLE

 Default query optimization class         (DFT_QUERYOPT) = 3
 Degree of parallelism                      (DFT_DEGREE) = 1
 Continue upon arithmetic exceptions   (DFT_SQLMATHWARN) = NO
 Default refresh age                  (DFT_REFRESH_AGE) = 0
 Default maintained table types for opt (DFT_MTTB_TYPES) = SYSTEM
 Number of frequent values retained     (NUM_FREQVALUES) = 10
 Number of quantiles retained            (NUM_QUANTILES) = 20

 Backup pending                                        = NO

 Database is consistent                                = YES
 Rollforward pending                                   = NO
 Restore pending                                       = NO

 Multi-page file allocation enabled                    = NO

 Log retain for recovery status                        = NO
 User exit for logging status                          = NO

 Data Links Token Expiry Interval (sec)      (DL_EXPINT) = 60
 Data Links Write Token Init Expiry Intvl(DL_WT_IEXPINT) = 60
 Data Links Number of Copies             (DL_NUM_COPIES) = 1
 Data Links Time after Drop (days)        (DL_TIME_DROP) = 1
 Data Links Token in Uppercase                (DL_UPPER) = NO
 Data Links Token Algorithm                   (DL_TOKEN) = MAC0

 Database heap (4KB)                            (DBHEAP) = 4800
 Size of database shared memory (4KB)  (DATABASE_MEMORY) = AUTOMATIC
 Catalog cache size (4KB)              (CATALOGCACHE_SZ) = 1024
 Log buffer size (4KB)                        (LOGBUFSZ) = 512
 Utilities heap size (4KB)                (UTIL_HEAP_SZ) = 10000
 Buffer pool size (pages)                     (BUFFPAGE) = 1000
 Extended storage segments size (4KB)    (ESTORE_SEG_SZ) = 16000
 Number of extended storage segments   (NUM_ESTORE_SEGS) = 0
 Max storage for lock list (4KB)              (LOCKLIST) = 20000

 Max size of appl. group mem set (4KB) (APPGROUP_MEM_SZ) = 30000
 Percent of mem for appl. group heap   (GROUPHEAP_RATIO) = 70
 Max appl. control heap size (4KB)     (APP_CTL_HEAP_SZ) = 2048

 Sort heap thres for shared sorts (4KB) (SHEAPTHRES_SHR) = (SHEAPTHRES)
```

```
 Sort list heap (4KB)                         (SORTHEAP) = 2048
 SQL statement heap (4KB)                     (STMTHEAP) = 8192
 Default application heap (4KB)             (APPLHEAPSZ) = 8192
 Package cache size (4KB)                   (PCKCACHESZ) = (MAXAPPLS*8)
 Statistics heap size (4KB)               (STAT_HEAP_SZ) = 20000

 Interval for checking deadlock (ms)         (DLCHKTIME) = 10000
 Percent. of lock lists per application       (MAXLOCKS) = 30
 Lock timeout (sec)                        (LOCKTIMEOUT) = 300

 Changed pages threshold                 (CHNGPGS_THRESH) = 30
 Number of asynchronous page cleaners    (NUM_IOCLEANERS) = 16
 Number of I/O servers                    (NUM_IOSERVERS) = 18
 Index sort flag                             (INDEXSORT) = YES
 Sequential detect flag                      (SEQDETECT) = YES
 Default prefetch size (pages)         (DFT_PREFETCH_SZ) = 32

 Track modified pages                         (TRACKMOD) = OFF

 Default number of containers                            = 1
 Default tablespace extentsize (pages)   (DFT_EXTENT_SZ) = 32

 Max number of active applications            (MAXAPPLS) = 5000
 Average number of active applications       (AVG_APPLS) = 5
 Max DB files open per application            (MAXFILOP) = 500

 Log file size (4KB)                          (LOGFILSIZ) = 20000
 Number of primary log files                 (LOGPRIMARY) = 20
 Number of secondary log files                (LOGSECOND) = 10
 Changed path to log files                   (NEWLOGPATH) =
 Path to log files                                       =
/log_v2/NODE0000/
 Overflow log path                       (OVERFLOWLOGPATH) =
 Mirror log path                           (MIRRORLOGPATH) =
 First active log file                                   =
 Block log on disk full                 (BLK_LOG_DSK_FUL) = NO
 Percent of max active log space by transaction(MAX_LOG) = 0
 Num. of active log files for 1 active UOW(NUM_LOG_SPAN) = 0

 Group commit count                           (MINCOMMIT) = 3
 Percent log file reclaimed before soft chckpt (SOFTMAX) = 750
 Log retain for recovery enabled              (LOGRETAIN) = OFF
 User exit for logging enabled                (USEREXIT) = OFF

 HADR database role                                      = STANDARD
 HADR local host name                   (HADR_LOCAL_HOST) =
 HADR local service name                  (HADR_LOCAL_SVC) =
 HADR remote host name                 (HADR_REMOTE_HOST) =
 HADR remote service name               (HADR_REMOTE_SVC) =
 HADR instance name of remote server   (HADR_REMOTE_INST) =
 HADR timeout value                       (HADR_TIMEOUT) = 120
 HADR log write synchronization mode     (HADR_SYNCMODE) = NEARSYNC

 First log archive method                 (LOGARCHMETH1) = OFF
 Options for logarchmeth1                  (LOGARCHOPT1) =
 Second log archive method                (LOGARCHMETH2) = OFF
 Options for logarchmeth2                  (LOGARCHOPT2) =
 Failover log archive path                (FAILARCHPATH) =
 Number of log archive retries on error    (NUMARCHRETRY) = 5
 Log archive retry Delay (secs)          (ARCHRETRYDELAY) = 20
 Vendor options                               (VENDOROPT) =

 Auto restart enabled                      (AUTORESTART) = ON
 Index re-creation time and redo index build  (INDEXREC) = RESTART
```

```
 Log pages during index build              (LOGINDEXBUILD) = OFF
 Default number of loadrec sessions     (DFT_LOADREC_SES) = 10
 Number of database backups to retain    (NUM_DB_BACKUPS) = 12
 Recovery history retention (days)      (REC_HIS_RETENTN) = 366

 TSM management class                     (TSM_MGMTCLASS) =
 TSM node name                             (TSM_NODENAME) =
 TSM owner                                    (TSM_OWNER) =
 TSM password                              (TSM_PASSWORD) =

 Automatic maintenance                        (AUTO_MAINT) = OFF
   Automatic database backup               (AUTO_DB_BACKUP) = OFF
   Automatic table maintenance             (AUTO_TBL_MAINT) = OFF
     Automatic runstats                     (AUTO_RUNSTATS) = OFF
     Automatic statistics profiling      (AUTO_STATS_PROF) = OFF
       Automatic profile updates           (AUTO_PROF_UPD) = OFF
     Automatic reorganization                 (AUTO_REORG) = OFF
```

Database Manager Configuration

```
 Node type = Enterprise Server Edition with local and remote clients

 Database manager configuration release level            = 0x0a00

 CPU speed (millisec/instruction)             (CPUSPEED) = 5.235149e-07
 Communications bandwidth (MB/sec)      (COMM_BANDWIDTH) = 1.000000e+02

 Max number of concurrently active databases     (NUMDB) = 8
 Data Links support                          (DATALINKS) = NO
 Federated Database System Support           (FEDERATED) = NO
 Transaction processor monitor name        (TP_MON_NAME) =

 Default charge-back account           (DFT_ACCOUNT_STR) =

 Java Development Kit installation path       (JDK_PATH) =
/usr/java14_64

 Diagnostic error capture level              (DIAGLEVEL) = 3
 Notify Level                              (NOTIFYLEVEL) = 3
 Diagnostic data directory path               (DIAGPATH) =
/home/v8inst2/sqllib/db2dump

 Default database monitor switches
   Buffer pool                          (DFT_MON_BUFPOOL) = OFF
   Lock                                    (DFT_MON_LOCK) = OFF
   Sort                                    (DFT_MON_SORT) = OFF
   Statement                               (DFT_MON_STMT) = OFF
   Table                                  (DFT_MON_TABLE) = OFF
   Timestamp                         (DFT_MON_TIMESTAMP) = ON
   Unit of work                             (DFT_MON_UOW) = OFF
 Monitor health of instance and databases   (HEALTH_MON) = ON

 SYSADM group name                          (SYSADM_GROUP) = V8GRP1
 SYSCTRL group name                        (SYSCTRL_GROUP) =
 SYSMAINT group name                      (SYSMAINT_GROUP) =
 SYSMON group name                          (SYSMON_GROUP) =

 Client Userid-Password Plugin           (CLNT_PW_PLUGIN) =
 Client Kerberos Plugin                 (CLNT_KRB_PLUGIN) =
 Group Plugin                              (GROUP_PLUGIN) =
 GSS Plugin for Local Authorization    (LOCAL_GSSPLUGIN) =
```

```
 Server Plugin Mode                      (SRV_PLUGIN_MODE) = UNFENCED
 Server List of GSS Plugins      (SRVCON_GSSPLUGIN_LIST) =
 Server Userid-Password Plugin        (SRVCON_PW_PLUGIN) =
 Server Connection Authentication          (SRVCON_AUTH) =
NOT_SPECIFIED
 Database manager authentication         (AUTHENTICATION) = SERVER
 Cataloging allowed without authority   (CATALOG_NOAUTH) = NO
 Trust all clients                      (TRUST_ALLCLNTS) = YES
 Trusted client authentication          (TRUST_CLNTAUTH) = CLIENT
 Bypass federated authentication            (FED_NOAUTH) = NO

 Default database path                       (DFTDBPATH) =
/home/v8inst2

 Database monitor heap size (4KB)          (MON_HEAP_SZ) = 8192
 Java Virtual Machine heap size (4KB)     (JAVA_HEAP_SZ) = 2048
 Audit buffer size (4KB)                  (AUDIT_BUF_SZ) = 0
 Size of instance shared memory (4KB)  (INSTANCE_MEMORY) = AUTOMATIC
 Backup buffer default size (4KB)            (BACKBUFSZ) = 1024
 Restore buffer default size (4KB)           (RESTBUFSZ) = 1024

 Sort heap threshold (4KB)                  (SHEAPTHRES) = 40000

 Directory cache support                     (DIR_CACHE) = YES

 Application support layer heap size (4KB)   (ASLHEAPSZ) = 15
 Max requester I/O block size (bytes)         (RQRIOBLK) = 32767
 Query heap size (4KB)                   (QUERY_HEAP_SZ) = 1000

 Workload impact by throttled utilities(UTIL_IMPACT_LIM) = 10

 Priority of agents                           (AGENTPRI) = SYSTEM
 Max number of existing agents               (MAXAGENTS) = 5000
 Agent pool size                        (NUM_POOLAGENTS) = 200
 Initial number of agents in pool       (NUM_INITAGENTS) = 200
 Max number of coordinating agents     (MAX_COORDAGENTS) = (MAXAGENTS -
NUM_INITAGENTS)
 Max no. of concurrent coordinating agents  (MAXCAGENTS) =
MAX_COORDAGENTS
 Max number of client connections      (MAX_CONNECTIONS) =
MAX_COORDAGENTS

 Keep fenced process                         (KEEPFENCED) = YES
 Number of pooled fenced processes          (FENCED_POOL) =
MAX_COORDAGENTS
 Initial number of fenced processes     (NUM_INITFENCED) = 0

 Index re-creation time and redo index build  (INDEXREC) = RESTART

 Transaction manager database name           (TM_DATABASE) = 1ST_CONN
 Transaction resync interval (sec)     (RESYNC_INTERVAL) = 180

 SPM name                                       (SPM_NAME) = el9_92_4
 SPM log size                           (SPM_LOG_FILE_SZ) = 256
 SPM resync agent limit                  (SPM_MAX_RESYNC) = 20
 SPM log path                              (SPM_LOG_PATH) =

 TCP/IP Service name                            (SVCENAME) = 58122
 Discovery mode                                 (DISCOVER) = SEARCH
 Discover server instance                (DISCOVER_INST) = ENABLE

 Maximum query degree of parallelism   (MAX_QUERYDEGREE) = 1
 Enable intra-partition parallelism     (INTRA_PARALLEL) = NO
```

```
 No. of int. communication buffers(4KB)(FCM_NUM_BUFFERS) = 4096
 Number of FCM request blocks              (FCM_NUM_RQB) = AUTOMATIC
 Number of FCM connection entries      (FCM_NUM_CONNECT) = AUTOMATIC
 Number of FCM message anchors         (FCM_NUM_ANCHORS) = AUTOMATIC

 Node connection elapse time (sec)         (CONN_ELAPSE) = 10
 Max number of node connection retries (MAX_CONNRETRIES) = 5
 Max time difference between nodes (min) (MAX_TIME_DIFF) = 60

 db2start/db2stop timeout (min)        (START_STOP_TIME) = 10

DB2SYSTEM=el9-92-62
DB2ADMINSERVER=v8asusr1
DB2_USE_ALTERNATE_PAGE_CLEANING=YES
DB2_PIPELINED_PLANS=ON
DB2_KEEPTABLELOCK=ON
DB2_INTERESTING_KEYS=ON
DB2_EVALUNCOMMITTED=TRUE
DB2TCPCONNMGRS=8
DB2_STRIPED_CONTAINERS=ON
DB2ASSUMEUPDATE=ON
DB2_CORRELATED_PREDICATES=yes
DB2_MMAP_WRITE=off
DB2_MMAP_READ=off
DB2_RR_TO_RS=YES
DB2COMM=tcpip
DB2CODEPAGE=1252
DB2AUTOSTART=TRUE
```

Buffer Pool Configuration

```
BPNAME
BUFFERPOOLID DBPGNAME
NPAGES      PAGESIZE    ESTORE NUMBLOCKPAGES BLOCKSIZE   NGNAME
------------------------------------------------------------------------
------------------------------------------------------------ ------------
------------------------------------------------------------------------
------------------------------------------------------------ ----------- -
--------- ------ ------------ ---------- --------------------------
------------------------------------------------------------------------
------------------------------
IBMDEFAULTBP
1 -
70000        4096 N                  0           0 -
BUF_16K
2 -
45000       16384 N                  0           0 -
BUF_IDX16K
3 -
80000       16384 N                  0           0 -
BUF_32K
4 -
5000        32768 N                  0           0 -

  4 record(s) selected.
```

Tablespace Configuration

```
Tablespace ID                           = 0
Name                                    = SYSCATSPACE
Type                                    = System managed space
Contents                                = Any data
State                                   = 0x0000
  Detailed explanation:
    Normal
Total pages                             = 291005
Useable pages                           = 291005
Used pages                              = 291005
Free pages                              = Not applicable
High water mark (pages)                 = Not applicable
Page size (bytes)                       = 4096
Extent size (pages)                     = 32
Prefetch size (pages)                   = 128
Number of containers                    = 1

Tablespace ID                           = 1
Name                                    = TEMPSPACE1
Type                                    = System managed space
Contents                                = System Temporary data
State                                   = 0x0000
  Detailed explanation:
    Normal
Total pages                             = 1
Useable pages                           = 1
Used pages                              = 1
Free pages                              = Not applicable
High water mark (pages)                 = Not applicable
Page size (bytes)                       = 4096
Extent size (pages)                     = 32
Prefetch size (pages)                   = 128
Number of containers                    = 1

Tablespace ID                           = 2
Name                                    = USERSPACE1
Type                                    = System managed space
Contents                                = Any data
State                                   = 0x0000
  Detailed explanation:
    Normal
Total pages                             = 38583
Useable pages                           = 38583
Used pages                              = 38583
Free pages                              = Not applicable
High water mark (pages)                 = Not applicable
Page size (bytes)                       = 4096
Extent size (pages)                     = 32
Prefetch size (pages)                   = 128
Number of containers                    = 1

Tablespace ID                           = 3
Name                                    = TBS_4K
Type                                    = Database managed space
Contents                                = Any data
State                                   = 0x0000
  Detailed explanation:
    Normal
Total pages                             = 16580608
Useable pages                           = 16580576
Used pages                              = 14023552
```

```
Free pages                              = 2557024
High water mark (pages)                 = 15712096
Page size (bytes)                       = 4096
Extent size (pages)                     = 32
Prefetch size (pages)                   = 128
Number of containers                    = 1

Tablespace ID                           = 4
Name                                    = TBS_16K
Type                                    = Database managed space
Contents                                = Any data
State                                   = 0x0000
  Detailed explanation:
    Normal
Total pages                             = 7077888
Useable pages                           = 7077792
Used pages                              = 4368448
Free pages                              = 2709344
High water mark (pages)                 = 6064224
Page size (bytes)                       = 16384
Extent size (pages)                     = 32
Prefetch size (pages)                   = 128
Number of containers                    = 3

Tablespace ID                           = 5
Name                                    = TBS_32K
Type                                    = Database managed space
Contents                                = Any data
State                                   = 0x0000
  Detailed explanation:
    Normal
Total pages                             = 655360
Useable pages                           = 655232
Used pages                              = 216096
Free pages                              = 439136
High water mark (pages)                 = 307872
Page size (bytes)                       = 32768
Extent size (pages)                     = 32
Prefetch size (pages)                   = 128
Number of containers                    = 4

Tablespace ID                           = 6
Name                                    = TEMP_16K
Type                                    = System managed space
Contents                                = System Temporary data
State                                   = 0x0000
  Detailed explanation:
    Normal
Total pages                             = 1
Useable pages                           = 1
Used pages                              = 1
Free pages                              = Not applicable
High water mark (pages)                 = Not applicable
Page size (bytes)                       = 16384
Extent size (pages)                     = 32
Prefetch size (pages)                   = 128
Number of containers                    = 1

Tablespace ID                           = 7
Name                                    = TEMP_32K
Type                                    = System managed space
Contents                                = System Temporary data
State                                   = 0x0000
  Detailed explanation:
```

```
     Normal
 Total pages                             = 1
 Useable pages                           = 1
 Used pages                              = 1
 Free pages                              = Not applicable
 High water mark (pages)                 = Not applicable
 Page size (bytes)                       = 32768
 Extent size (pages)                     = 64
 Prefetch size (pages)                   = 256
 Number of containers                    = 1

 Tablespace ID                           = 8
 Name                                    = TBS_IDX
 Type                                    = Database managed space
 Contents                                = Any data
 State                                   = 0x0000
   Detailed explanation:
     Normal
 Total pages                             = 7077888
 Useable pages                           = 7077864
 Used pages                              = 4409696
 Free pages                              = 2668168
 High water mark (pages)                 = 4476800
 Page size (bytes)                       = 16384
 Extent size (pages)                     = 8
 Prefetch size (pages)                   = 128
 Number of containers                    = 3

DB20000I  The SQL command completed successfully.
```

Tablespace Containers Configuration

```
            Tablespace Containers for Tablespace 1

 Container ID                            = 0
 Name                                    =
/home/v8inst2/dbvols/fs_v2/v2_1_tmp4k
 Type                                    = Path
 Total pages                             = 1
 Useable pages                           = 1
 Accessible                              = Yes


            Tablespace Containers for Tablespace 2

 Container ID                            = 0
 Name                                    =
/home/v8inst2/dbvols/fs_v2/v2_2_usr4k
 Type                                    = Path
 Total pages                             = 38583
 Useable pages                           = 38583
 Accessible                              = Yes


            Tablespace Containers for Tablespace 3

 Container ID                            = 0
 Name                                    = /home/v8inst2/dbvols/rv2_3_4k
 Type                                    = Disk
 Total pages                             = 16580608
 Useable pages                           = 16580576
```

```
 Accessible                              = Yes


            Tablespace Containers for Tablespace 4

 Container ID                            = 0
 Name                                    = /home/v8inst2/dbvols/rv2_41_16k
 Type                                    = Disk
 Total pages                             = 2359296
 Useable pages                           = 2359264
 Accessible                              = Yes
 Container ID                            = 1
 Name                                    = /home/v8inst2/dbvols/rv2_42_16k
 Type                                    = Disk
 Total pages                             = 2359296
 Useable pages                           = 2359264
 Accessible                              = Yes
 Container ID                            = 2
 Name                                    = /home/v8inst2/dbvols/rv2_43_16k
 Type                                    = Disk
 Total pages                             = 2359296
 Useable pages                           = 2359264
 Accessible                              = Yes


            Tablespace Containers for Tablespace 5

 Container ID                            = 0
 Name                                    = /home/v8inst2/dbvols/rv2_51_32k
 Type                                    = Disk
 Total pages                             = 163840
 Useable pages                           = 163808
 Accessible                              = Yes
 Container ID                            = 1
 Name                                    = /home/v8inst2/dbvols/rv2_52_32k
 Type                                    = Disk
 Total pages                             = 163840
 Useable pages                           = 163808
 Accessible                              = Yes
 Container ID                            = 2
 Name                                    = /home/v8inst2/dbvols/rv2_53_32k
 Type                                    = Disk
 Total pages                             = 163840
 Useable pages                           = 163808
 Accessible                              = Yes
 Container ID                            = 3
 Name                                    = /home/v8inst2/dbvols/rv2_54_32k
 Type                                    = Disk
 Total pages                             = 163840
 Useable pages                           = 163808
 Accessible                              = Yes


            Tablespace Containers for Tablespace 6

 Container ID                            = 0
 Name                                    =
/home/v8inst2/dbvols/fs_v2/v2_6_tmp16k
 Type                                    = Path
 Total pages                             = 1
 Useable pages                           = 1
 Accessible                              = Yes
```

```
            Tablespace Containers for Tablespace 7

 Container ID                            = 0
 Name                                    =
/home/v8inst2/dbvols/fs_v2/v2_7_tmp32k
 Type                                    = Path
 Total pages                             = 1
 Useable pages                           = 1
 Accessible                              = Yes


            Tablespace Containers for Tablespace 8

 Container ID                            = 0
 Name                                    = /home/v8inst2/dbvols/rv2_81_16k
 Type                                    = Disk
 Total pages                             = 2359296
 Useable pages                           = 2359288
 Accessible                              = Yes
 Container ID                            = 1
 Name                                    = /home/v8inst2/dbvols/rv2_82_16k
 Type                                    = Disk
 Total pages                             = 2359296
 Useable pages                           = 2359288
 Accessible                              = Yes
 Container ID                            = 2
 Name                                    = /home/v8inst2/dbvols/rv2_83_16k
 Type                                    = Disk
 Total pages                             = 2359296
 Useable pages                           = 2359288
 Accessible                              = Yes
```

Appendix I: Additional Information

These Web sites provide useful reference materials to supplement the information contained within this document:

- IBM eServer x366 Product Information:
  ibm.com/servers/eserver/xseries/x366.html

- IBM eServer x460 Product Information:
  ibm.com/servers/eserver/xseries/x460.html

- IBM TotalStorage DS4300 Product Information:
  ibm.com/servers/storage/disk/ds4000/ds4300/index.html

About the author

Mark Trbojevic
Advisory Software Engineer

Mark Trbojevic is an advisory software engineer with the ISV Business Strategy and Enablement Group based in Beaverton, Oregon. Mark has more than 19 years of experience in the computing industry and has been with IBM for seven years. He has a background in developing and designing bespoke customer relationship management (CRM) software solutions for blue-chip companies. Mark has been supporting Siebel Systems solutions on pSeries hardware for more than five years and has been a key contributor during all industry-leading Siebel benchmarks that have been published for the pSeries and xSeries platforms by IBM and Siebel during this time. He also specializes in load balancing and IBM High Availability Cluster Multiprocessing (HACMP™) software.

Trademarks and Special Notices

References in this document to IBM products or services do not imply that IBM intends to make them available in every country.

The following terms are trademarks of International Business Machines Corporation in the United States, other countries, or both:

IBM	eServer	AIX	DB2
The IBM logo	xSeries	AIX 5L	DB2 Universal Database
ibm.com	pSeries	X-Architecture	TotalStorage
Redbooks	Netfinity	BladeCenter	HACMP
POWER4			

Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

Other company, product or service names may be trademarks or service marks of others.

Information is provided "AS IS" without warranty of any kind.

Information concerning non-IBM products was obtained from a supplier of these products, published announcement material, or other publicly available sources and does not constitute an endorsement of such products by IBM. Sources for non-IBM list prices and performance numbers are taken from publicly available information, including vendor announcements and vendor worldwide homepages. IBM has not tested these products and cannot confirm the accuracy of performance, capability, or any other claims related to non-IBM products. Questions on the capability of non-IBM products should be addressed to the supplier of those products.