Lawson M3 System Foundation 7.1 using IBM Technology for Java on System i

IBM System i5

System i ERP, LawsonTeam

January 2007

Introduction 3
Statement of Approval 4
Software 5
PTFs/Fixes required. 6
Database 7
Business Flow 8
Benchmark Methodology 9
Warm-up/Ramp-up Phase 10
Tuning: 11
IBM Technology for Java: 12
System Configuration: 13
Results 14
SP15 vs. M3 System Foundation 7.1 14
Classic JVM using JIT vs. IT4J 16
M3 WorkPlace on WebSphere 5.1 vs. WebSphere 6.1 18
Typical Customer Scenario 20
Scaling Results 22
Summary 24
Trademarks and Disclaimers 25

Introduction

This paper describes testing that was done to test flexibility, integrability, and scalability of the new Lawson M3 System Foundation 7.1 using the new IBM Technology for Java Virtual Machine, or IT4J, now available in V5R4.

Lawson M3 is a supplier of collaboration software which focuses on the manufacturing, maintenance, and distribution industries and servers many customers around the world.
The Lawson M3 ERP solution tested here is a Java™-based application which runs on System i servers.

The new Lawson M3 System Foundation 7.1 introduces several new key features. These include:

- A new foundation which now is decoupled from the business logic
- Support for server side pooling of API connections
- Support for Data compression in API connections
- Support for the new IBM Technology for Java Virtual Machine now available in V5R4.
- Support JDK 1.5.
- Support for WebSphere 6.1 with M3 WorkPlace

To demonstrate the abilities of M3 System Foundation 7.1 and IT4J, a test scenario was designed to simulate a typical customer. A typical M3 customer is defined as a customer who processes transaction through an user interface into the M3 Business Engine. This process is the foundation for most transactions done in the M3 environment and what is projected that most customers' do. The work load run in this report tries to simulate this process as close as possible. The test scenario was run using the M3 One Box solution. The M3 One Box solution consists of two applications running simultaneously on the same system, "One Box". The test scenario was as follows: simulated users entered orders through M3 Workplace into M3 Business Engine, which includes full back-end processing to invoice the orders. The test team then ran the One Box scenario using a System i5 model 570 system. Several different test scenarios where run to show the performance and scalability of both M3 System Foundation 7.1 and IT4J. For more details on IT4J please refer to the IBM Technology for Java section of this report.

All tests achieved excellent results, including subsecond response times and at least 1,000 order lines processed per minute. Overall, the M3 System Foundation 7.1 achieved excellent scalability using IT4J on System i and had a much smaller memory footprint. When compared to previous M3 releases, which rely on Direct Execution for Java compilation, a significant performance gain can be seen as well.

This report is a demonstration of the capabilities of the System i5:

- The ability to run multiple complex workloads even on a small single processor system.
- The integration of i5/OS, DB2® UDB for System i5, Java, and WebSphere Application Server demonstrates the ability of i5 systems to support new application models.
- Advantages that new technology such as the IBM Technology for Java virtual machine can bring to an ISV application.
- Flexibility of System i5 to handle workloads on even small environments and grow as your business grows.

Statement of Approval

Lawson M3 has reviewed, verified, and approved results for their applications which are shown in this report.

- Lars Strandner, Lead Architect Platforms, Lawson M3 R&D, lars.strandner@se.lawson.com

Software

Lawson M3 Software:

The tests described in this report used the latest versions of three Lawson M3 products:

M3 Business Engine 12.4.3 SP15 and System Foundation 7.1– Business Engine provides the foundation for Lawson's M3 business software. Although M3 Business Engine has its roots in the OS/400 and RPG platforms, the current version is built with Java technology, providing additional flexibility.

M3 Workplace 5.2.1 and 5.2.2– M3 Workplace is a role-based user interface that combines portal and Web technology with predefined work flows. These predefined work flows can be modified and combined with company-specific components and work flows. M3 Workplace can provide employees with fast, easy access to work-related information and applications, thereby potentially helping increase their productivity and efficiency. Workplace is built on top of IBM WebSphere Application Server V5.1.

IBM Software:

The tests described in this report used the following IBM software:

WebSphere Application Server Base v5.1.1.11 and v6.0.0.1 for OS/400
WebSphere Application Server provides a J2EE-compliant environment that allows you to manage server-side Java components. These Java components can add complex business logic and dynamic functions to static HTML Web pages and to standalone Java application clients.

i5/OS V5R4
IBM System i5 servers run many applications that clients have today while adapting quickly and easily to accommodate applications that may be needed in the future. The primary operating system, the i5/OS, due to its breadth and richness of integrated functions, offers great versatility and operational efficiency. It does this by natively and concurrently supporting applications built in RPG, COBOL, C, C++, Java, WebSphere and Domino.

i5/OS is fully integrated right out of the box. This means that relational database, communication and networking capabilities, online help, Web enablement technologies, easy enterprise management and much more are fully integrated into the operating system and the machine. The user communicates with all components of i5/OS using a single command language or a simple unified graphical user interface (GUI).

PTFs/Fixes required.

The following group PTFs were applied on the systems:

PTF Group	Level	Product
SF99504	4	DB2 UDB for i5/OS
SF99291	3	Java
SF99308	4	WebSphere App Server v5.1 (Base)
SF99312	4	WebSphere App Server v6.0
SF99114	3	IBM HTTP Server for i5/OS

In addition to the fixes above one fix for Lawson M3a was required, for SP15 only, for the class below:

MvxPool.class - Error alert - 20050519/0184

This Lawson M3 change corrects a memory leak that was causing degraded performance.

Database

The IBM test team configured the M3 Business Engine database with the following data:

- Customers: 200,017 (OCUSMA table)
- Items: 100,856 (MITMAS table)
- Customer locations: 2,000

This database contained 3,045 physical files and 8,012 logical files; the total size of the database library (not including journals and journal receivers) was 5.47 GB at the beginning of the run. In the scalability tests, the database library grew to 15.42 GB on the 4-way. The database increased by a smaller amount on the 2-way and 1-way.

The test M3 Workplace database was created with the standard Workplace installation program, and then populated with 2,001 additional users. During the benchmark runs, each virtual user logged in as one of these users. The Workplace tree was populated with M3 programs by importing the MOVEX_DEMO_GB.xml file included in the Workplace installation.

The Workplace database contained 3 tables, 19 indexes, and 305 stored procedures. The total size of the database library was 92MB at the beginning of the run. The Workplace database did not change in size significantly during the course of a run.

Business Flow

The IBM test team used a load generation tool to simulate users accessing M3 Workplace through a Web browser. Each user logged to M3 Workplace in and then entered a series of orders through M3 Net Extension, which was connected to M3 Business Engine on the same server. Each order consisted of five order lines. At the end of the run, each virtual user logged out.

M3 Workplace was running inside a single instance of WebSphere Application Server V5.1. Users connected to an Apache server running on the same server, which then passed the requests to WebSphere.

The test team configured M3 Business Engine to fully process each order after it was entered, including the following steps:

- Checking agreements
- Checking item aliases
- Checking buying groups
- Checking order charges
- Checking assortments
- Automatic shipment connection
- Automatic route selection
- Automatic loading dock selection
- Price lookup through multiple levels

The M3 AutoJobs perform all of these steps through background batch processing; therefore, users do not have to wait for these steps to complete before continuing with the next order. The test team tuned the M3 AutoJobs so that they completed processing orders at about the same rate that new orders were entered. This tuning helped ensure that the measured throughput could be maintained over an extended period of time. Because the AutoJobs ran in the background, the system could handle periods of higher peak utilization; during such periods, the AutoJobs would temporarily fall behind but would be able to catch up later during periods of lower utilization.

Figure 1 shows an overview of the major system components involved in the test.

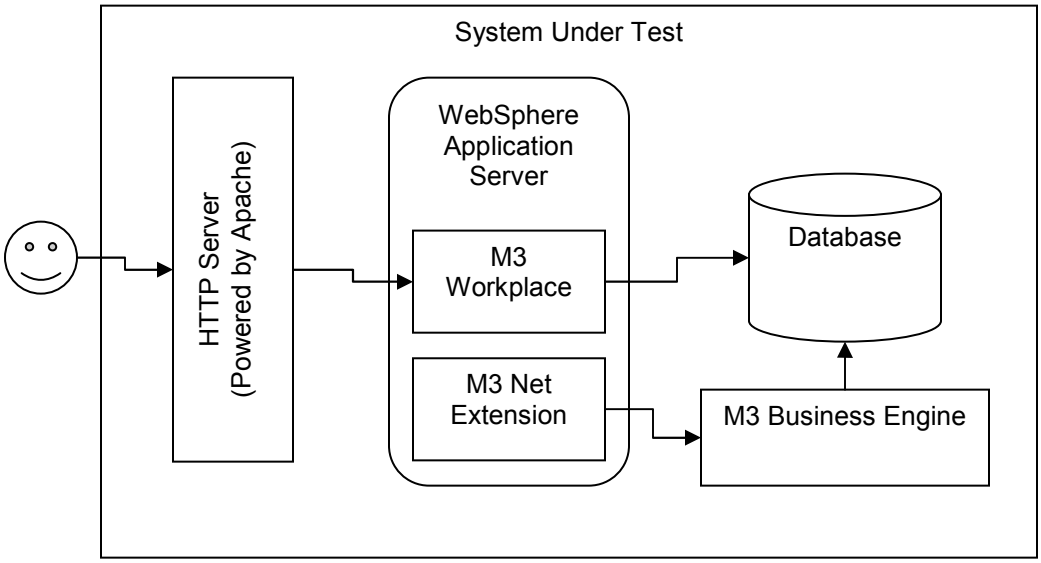


Figure 1. Major system components in System i5 Lawson M3 benchmark test

Benchmark Methodology

The IBM test team configured the load generation tool to simulate a small number of virtual users entering orders at a fast rate. Each user was configured to enter 900 orders during the run. To maximize throughput, a series of practice runs were made on each server to choose the optimal number of users for that server.

After logging in, the user performed an order entry operation consisting of seven steps:

1. Create a new order
2. Enter order line 1
3. Enter order line 2
4. Enter order line 3
5. Enter order line 4
6. Enter order line 5
7. Close the order

The virtual users were configured to wait an average of one second between each step. This time period—the *think time*—includes the time required for the browser to render the page as well as the time required for the user to think about what to do next, to type in new information, and to submit the form back to the server. One-second think time is typically faster than a real-world user would normally perform each step. Therefore, the small number of virtual users in the test may represent a much larger number of real-world users.

Because full order processing consists of both an interactive and a batch component, the primary metric for this benchmark is *number of invoiced order lines per hour*, rather than the number of order lines entered per hour. An invoiced order line has completed all of the interactive and batch processing required for that order. To measure the number of invoiced order lines, the OOLINE table in the M3 database was monitored for order lines with a status of “77,” indicating that processing was complete for that line.

The number of invoiced order lines was counted once every minute throughout the benchmark test. Once the run was complete, the number of invoiced order lines per hour over a particular measurement interval was calculated from this data. The measurement interval was defined to be a 90 minute period beginning 15 minutes after the run started. This definition gives a more accurate view of the relative performance when comparing systems with significantly different performance characteristics, since the ramp-up period for each configuration will be somewhat different, making comparisons difficult.

To demonstrate that the results were repeatable, each result consisted of two runs with identical parameters.

Warm-up/Ramp-up Phase

For each run that was done a warm-up and ramp-up phase was performed. The warm-up phase consisted of 1 virtual user which does 10 iterations of the above defined scenario. This ensures all jobs have started and allows the JVM to warm-up by ensuring all the classes that will be used are loaded and any JSP or servlets have been initialized. Thus the warm-up phase can help prevent any timeouts or virtual users failing, as a result of slow responses due to first touch situations, during the run time phase.

After the warm-up is completed the run is then started. We now enter into the ramp-up phase. The virtual users are not started at the same time. Instead one virtual user is started every 5 seconds. The time it takes for all virtual users to have started is referred to as its ramp-up time. Having all virtual users start at the same time can overwhelm the system and cause jobs to back up and time outs or other errors may occur. A large number of users all logging in at the same time is an unrealistic real life scenario. Thus using a ramp-up of the uses is not only more similar to a real work scenario it also avoids overwhelming the system under test.

Tuning:

- Class reloading was disabled in Workplace WebSphere profile. Class reloading is normally enabled during development but should be disabled (or set to a high reload interval) in production environments.

- The number of Web container threads used for Workplace WebSphere profile was changed to 15. (The default is 80.) This limited the number of requests processed concurrently by Workplace, reducing contention, and therefore improving the average response time experienced by users.

- Transport planning was turned off for the M3 Business Engine.

- The Minimum and Maximum Heap Sizes were tuned for each of the M3 JVMs to minimize overhead caused by garbage collection.

- Below are the subsystem settings that were used for each subsystem. These were used for all tests expect where noted.

```
# Supervisor
Supervisor_MaxJobs=0
Supervisor_MaxHeap=360m      - IT4J tests
Supervisor_MaxHeap=*NOMAX    - Classic JVM tests
Supervisor_MinHeap=24m       - SP15 tests both Classic JVM and IT4J
Supervisor_MinHeap=64m       - System Foundation 7.1 tests both Classic JVM
                               and IT4J

# Autojob
Auto_MaxJobs=45
Auto_MaxHeap=512m            - IT4J tests
Auto_MaxHeap=*NOMAX          - Classic JVM tests
Auto_MinHeap=256m

# Interactive
Interactive_MaxJobs=250
Interactive_MaxHeap=2048m    - IT4J tests
Interactive_MaxHeap=*NOMAX   - Classic JVM tests
Interactive_MinHeap=256m

# Batchjob
Batch_MaxJobs=15
Batch_MaxHeap=640m           - IT4J tests
Batch_MaxHeap=*NOMAX         - Classic JVM tests
Batch_MinHeap=24m

# Apijob
Api_MaxJobs=15
Api_MaxHeap=640m             - IT4J tests
Api_MaxHeap=*NOMAX           - Classic JVM tests
Api_MinHeap=24m
```

IBM Technology for Java:

With the M3 System Foundation 7.1 comes support for the new IBM Technology for Java virtual machine. This 32-bit JVM can have several advantages; one of the biggest can be reduced memory footprint. This section gives an overview of the new IBM Technology for Java virtual machine and how it is used in this report.

Overview of IT4J
As of V5R4 there is now a second JVM available on System i, referred to as IBM Technology for Java or IT4J. This new JVM is a 32-bit JVM, which currently supports JDK 1.5 only, runs inside of the Portable Application Solutions Environment or PASE. A 64-bit version of IT4J will be available in the future. The original System i 64-bit JVM, referred in this report as the Classic JVM, remains fully supported and unchanged. The Classic JVM supports JDK 1.5, JDK 1.4, and earlier releases. Also the Classic JVM is the default JVM used in V5R4. Some minor changes are needed in order to be able to use IT4J.

Advantages
There are several advantages to IT4J. One key advantage this report shows is reduced memory footprint with a 32-bit JVM like IT4J, versus a 64-bit JVM such as the Classic JVM.
Another advantage of IT4J is it is based on the standard IBM JVM that is used by other platforms. Thus tuning and behavior should be similar to other platforms.

Disadvantages
Since IT4J is a 32-bit JVM that means there is a much smaller limit on how large the heap can grow. The heap with IT4J can grow to approximately 3.25 GB. If an application or size of the system requires a JVM larger than this limit then the Classic JVM should be used. The Classic JVM has a significantly higher maximum heap limit and offers excellent single JVM scalability. However, for most small to medium applications/systems IT4J may be a viable option when you consider the memory savings.

Just in Time Compilation versus the Direct Execution Compilation
In this report we will compare the Just in Time Compiler or JIT to the Direct Execution, or DE, static compilation. DE has been available on System i for quite some time. DE allowed programs to be compiled and optimized ahead of time. In the past this had yielded much faster JVM startup times and runtime performance. However, in recent years the JIT has significantly overtaken DE when it comes to runtime performance, with startup times close to that of DE. DE is only available with the Classic JVM there is no support for DE in IT4J. However, the JIT is supported by both the Classic JVM and IT4J.

System Configuration:

To gauge the performance of Lawson M3 software on the System i the IBM test team configured a System i5 model 570 server with the following resources:

System i5 model 570-7747
Processors: 4
CPW rating: 16,000
Main storage: 32,768 MB
Disk: 78 arms, 15k rpm, 35GB capacity
Disk configuration: Single ASP with device parity protection
Disk IOP: Disks spread across twelve 571B IOPs
Network: 100Mbps Ethernet, full duplex

For the scaling tests each of the three measurement sets used a different number of processors. The three measurements sets used an identical configuration for disk, memory, network, operating system, and applications.

	1-way	2-way	4-way
Processors	1	2	4
CPW Rating	~4200	8400	16,000
Main Storage	32 GB	32 GB	32 GB
Disk	78 arms, 15k RPM, 35 GB capacity		
Disk configuration	One ASP with device parity protection.		
Disk IOP:	571B IOPs.		
Network	100Mbps Ethernet, full duplex		
Operating System	i5/OS V5R4		

All system configurations used security level 40.

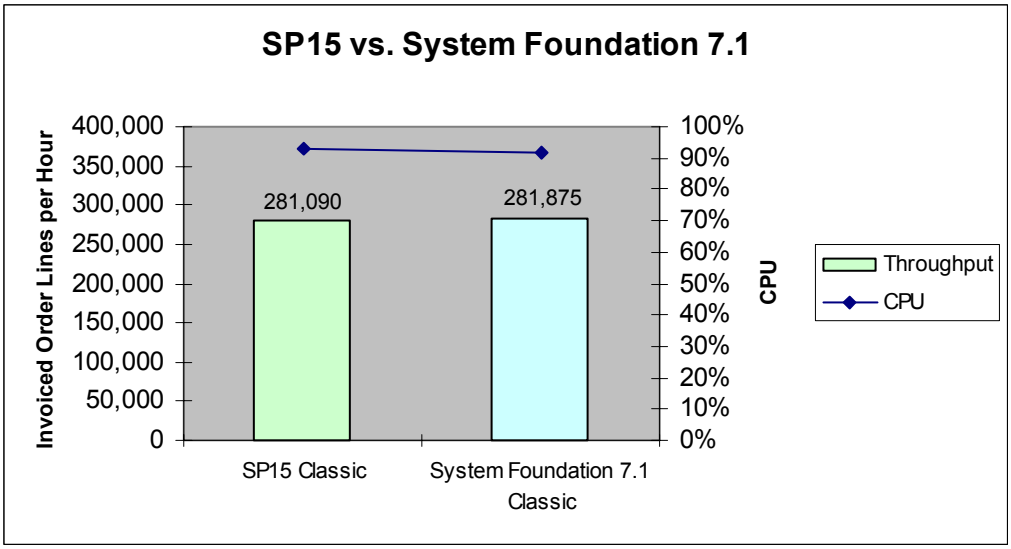
Results

The following sections describe the results of the testing that was done.

M3 SP15 vs. M3 System Foundation 7.1

The tests show that performance is slightly improved with the new M3 System Foundation 7.1. Hence, when moving to this new release of M3, you may be able to take advantage of the new features available in this release with no significant change in performance expected.
The tests also show that there is little to no change in memory usage with the M3 System Foundation 7.1.

Note: The Classic JVM using JIT was used for all tests in this section.



SP15 with the Classic JVM
M3 with SP15 invoiced 281,240order lines per hour in one run and 280,939order lines per hour in the second run. In both runs, 118 virtual users were used.

Response time. Table 1 shows the average response time for each Web browser transaction for M3 with SP15 V5R3. Response times do not include the average one-second think time for each transaction and they measure the entire length of the run, not just the 90 minute measurement interval.

Step	Number per Order	Average Response Time (seconds)		90th Percentile Response Time (seconds)	
		Run 1	Run 2	Run 1	Run 2
Create Order Head	1	0.128	0.128	0.15	0.15
Insert Order Line	5	0.072	0.073	0.079	0.079
Close Order	1	0.059	0.059	0.079	0.079

Table 1. Average response times for SP15 with the Classic JVM

M3 System Foundation 7.1 with the Classic JVM
M3 System Foundation 7.1 invoiced 282,145 order lines per hour in one run and 281,605 order lines per hour in the second run. In both runs, 118 virtual users were used.

Response time. Table 2 shows the average response time for each Web browser transaction on M3 System Foundation 7.1. Response times do not include the average one-second think time for each transaction and they measure the entire length of the run, not just the 90 minute measurement interval.

Step	Number per Order	Average Response Time (seconds)		90th Percentile Response Time (seconds)	
		Run 1	Run 2	Run 1	Run 2
Create Order Head	1	0.123	0.124	0.15	0.15
Insert Order Line	5	0.069	0.071	0.079	0.079
Close Order	1	0.056	0.058	0.079	0.079

***Table 2. Average response times for* M3 System Foundation 7.1**

Memory:
Table 3 below shows the memory usage per JVM for both M3 with SP15 and M3 System Foundation 7.1. As the data shows there is very little change in memory usage.

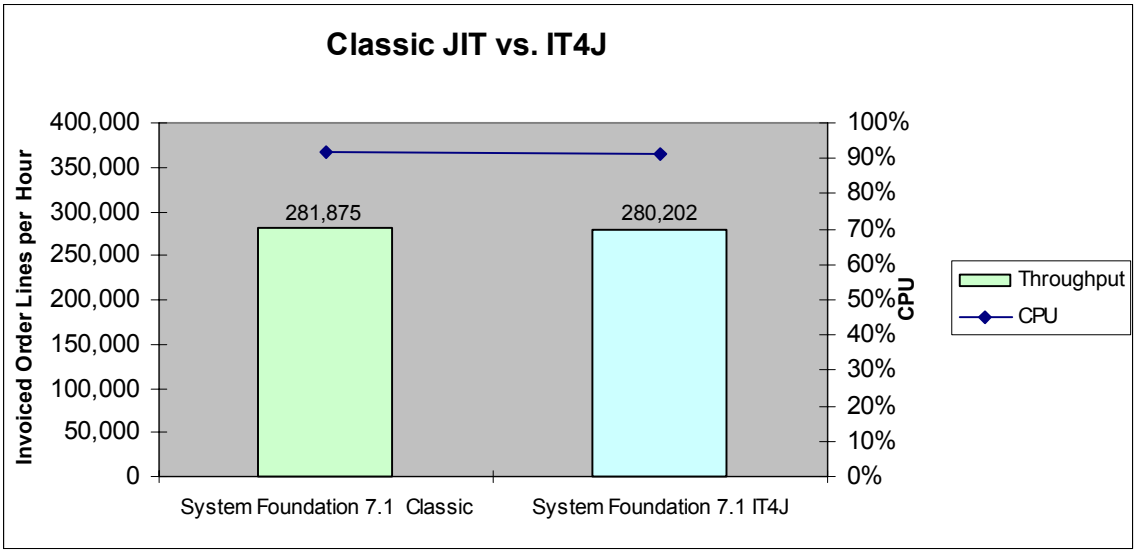
JVM Type	SP15	M3 System Foundation 7.1
Supervisor	70 MB	92 MB
Auto Job	277 MB	245 MB
Interactive	1300 MB	1284 MB

Table 3. Memory for Sp15 versus M3 System Foundation 7.1

CPU Utilization
The CPU(s) experienced 90 to 100% utilization for all of the tests in this section.

Classic JVM using JIT vs. IT4J

The tests show that performance is near equivalent with M3 System Foundation 7.1 using IT4J when compared to M3 System Foundation 7.1 using the Classic JVM with JIT. Thus an application like M3 System Foundation 7.1 can take advantage of the benefits of IT4J, such as reduced memory footprint, with no significant change in runtime performance. The memory benefits of IT4J will be shown in the next two result sections.



M3 System Foundation 7.1 with the Classic JVM using JIT
M3 System Foundation 7.1 with the Classic JVM invoiced 282,145 order lines per hour in one run and 281,605 order lines per hour in the second run. In both runs, 118 virtual users were used.

Response time. Table 4 shows the average response time for each Web browser transaction for M3 System Foundation 7.1 with the Classic JVM. Response times do not include the average one-second think time for each transaction and they measure the entire length of the run, not just the 90 minute measurement interval.

Step	Number per Order	Average Response Time (seconds)		90th Percentile Response Time (seconds)	
		Run 1	Run 2	Run 1	Run 2
Create Order Head	1	0.123	0.124	0.15	0.15
Insert Order Line	5	0.069	0.071	0.079	0.079
Close Order	1	0.056	0.058	0.079	0.079

Table 4. Average response times for M3 System Foundation 7.1 with the Classic JVM using JIT

M3 System Foundation 7.1 with IT4J
M3 System Foundation 7.1 with IT4J invoiced 280,183 order lines per hour in one run and 280,221 order lines per hour in the second run. In both runs, 118 virtual users were used.

Response time. Table 5 shows the average response time for each Web browser transaction on M3 System Foundation 7.1. Response times do not include the average one-second think time for each transaction and they measure the entire length of the run, not just the 90 minute measurement interval.

Step	Number per Order	Average Response Time (seconds)		90th Percentile Response Time (seconds)	
		Run 1	Run 2	Run 1	Run 2
Create Order Head	1	0.132	0.132	0.17	0.17
Insert Order Line	5	0.076	0.075	0.109	0.109
Close Order	1	0.065	0.065	0.099	0.099

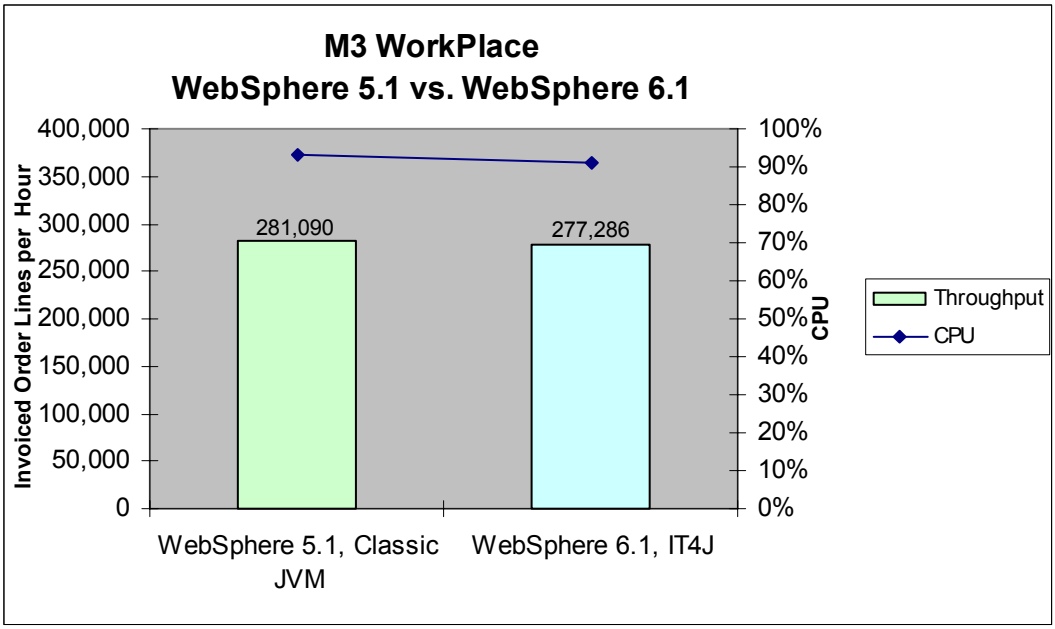
Table 5. Average response times for M3 System Foundation 7.1 with IT4J

CPU Utilization
The CPU(s) experienced 90 to 100% utilization for all of the tests in this section.

M3 WorkPlace on WebSphere 5.1 vs. WebSphere 6.1

M3 WorkPlace now supports WebSphere 6.1. In addition, WebSphere 6.1 supports both the Classic JVM and IT4J. The default is still the Classic JVM, but a customer can choose to enable WebSphere 6.1 to use IT4J instead. The tests below show what a typical customer can expect when moving from a WebSphere 5.1 based WorkPlace version, using the Classic JVM, to one that uses WebSphere 6.1 on IT4J. The results show that performance is near equivalent with M3 Workplace on WebSphere 5.1 when compared to M3 WorkPlace on WebSphere 6.1 using IT4J. However, memory footprint is significantly reduced when using WebSphere 6.1 with IT4J.

Note: The JIT was used for all tests below.



M3 WorkPlace with WebSphere 5.1 using the Classic JVM
M3 WorkPlace with WebSphere 5.1 invoiced 281,090 order lines per hour in one run and 281,605 order lines per hour in the second run. In both runs, 118 virtual users were used.

Response time. Table 6 shows the average response time for each Web browser transaction for M3 System Foundation 7.1 with the Classic JVM. Response times do not include the average one-second think time for each transaction and they measure the entire length of the run, not just the 90 minute measurement interval.

Step	Number per Order	Average Response Time (seconds)		90th Percentile Response Time (seconds)	
		Run 1	Run 2	Run 1	Run 2
Create Order Head	1	0.123	0.124	0.15	0.15
Insert Order Line	5	0.069	0.071	0.079	0.079
Close Order	1	0.056	0.058	0.079	0.079

Table 6. Average response times M3 WorkPlace with WebSphere 5.1 using the Classic JVM

M3 WorkPlace with WebSphere 6.1 using IT4J
M3 WorkPlace with WebSphere 6.1 invoiced 280,183 order lines per hour in one run and 280,221 order lines per hour in the second run. In both runs, 118 virtual users were used.

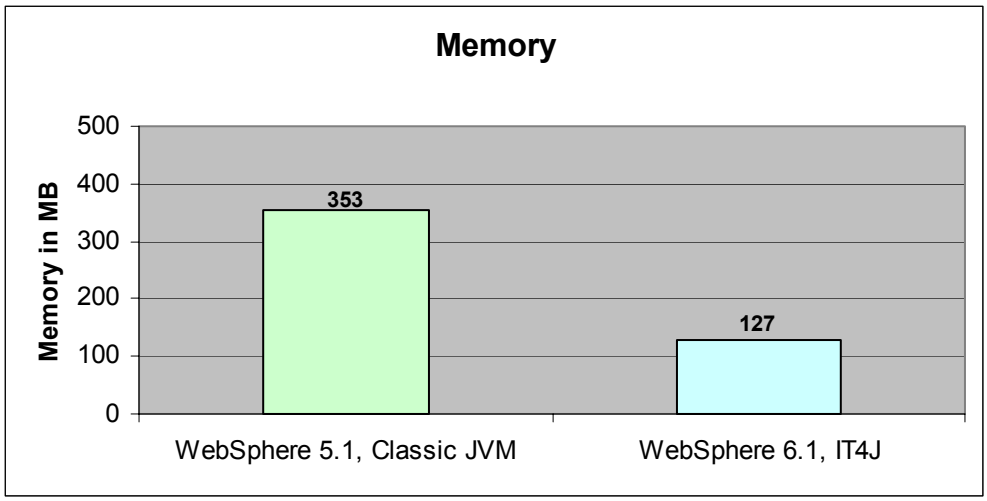
Response time. Table 7 shows the average response time for each Web browser transaction on M3 System Foundation 7.1. Response times do not include the average one-second think time for each transaction and they measure the entire length of the run, not just the 90 minute measurement interval.

Step	Number per Order	Average Response Time (seconds)		90th Percentile Response Time (seconds)	
		Run 1	Run 2	Run 1	Run 2
Create Order Head	1	0.132	0.132	0.17	0.17
Insert Order Line	5	0.076	0.075	0.109	0.109
Close Order	1	0.065	0.065	0.099	0.099

Table 7. Average response times for M3 WorkPlace with WebSphere 6.1 using IT4J

Memory:

The results below show that memory is significantly reduced when using WebSphere 6.1 with IT4J versus WebSphere 5.1 with the Classic JVM. This is one key advantage of a 32-bit JVM like IT4J.

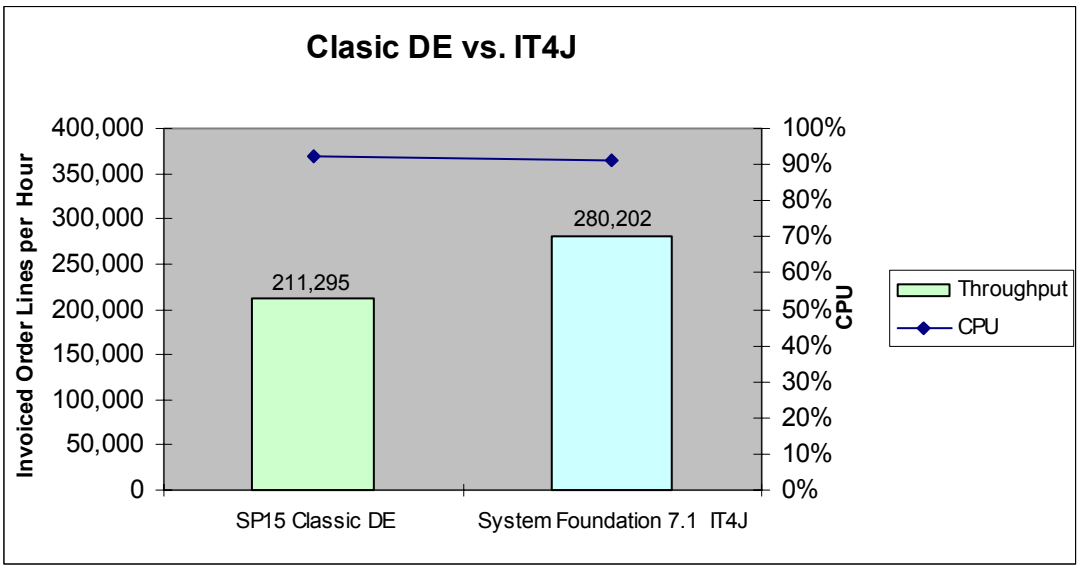


CPU Utilization
The CPU(s) experienced 90 to 100% utilization for all of the tests in this section.

Typical Customer Scenario

The results below show what a typical M3 customer could expect when moving from an older M3 release that is Direct Execution or DE based to the new System Foundation 7.1 running under IT4J. The tests show that a typical M3 customer can see a runtime performance improvement up to up 25% when moving to M3 System Foundation 7.1 using IT4J from an older M3 direct execution environment.
In addition memory footprint is significantly reduced when using IT4J.

Note: The gain shown below comes from moving from DE to JIT and not IT4J itself. As the previous section shows performance between the Classic JVM with JIT vs. IT4J is equivalent. However, M3 System Foundation 7.1 utilizes JIT by default and most previous M3 release use DE. Thus a typical M3 customer can expect to see a significant performance improvement when moving to M3 System Foundation 7.1.



SP15 with the Classic JVM using DE:
M3 SP15 using Classic JVM with DE invoiced 211,269 order lines per hour in one run and 211,321 order lines per hour in the second run. In both runs, 90 virtual users were used.

Response time. Table 8 shows the average response time for each Web browser transaction on M3 SP15 using the Classic JVM with DE. Response times do not include the average one-second think time for each transaction and they measure the entire length of the run, not just the 90 minute measurement interval.

Step	Number per Order	Average Response Time (seconds)		90th Percentile Response Time (seconds)	
		Run 1	Run 2	Run 1	Run 2
Create Order Head	1	0.147	0.148	0.18	0.18
Insert Order Line	5	0.09	0.09	0.099	0.099
Close Order	1	0.068	0.068	0.089	0.089

Table 8. Average response times for M3 SP15 using the Classic JVM with DE

M3 System Foundation 7.1 using IT4J
M3 System Foundation 7.1 with IT4J invoiced 280,183 order lines per hour in one run and 280,221 order lines per hour in the second run. In both runs, 118 virtual users were used.

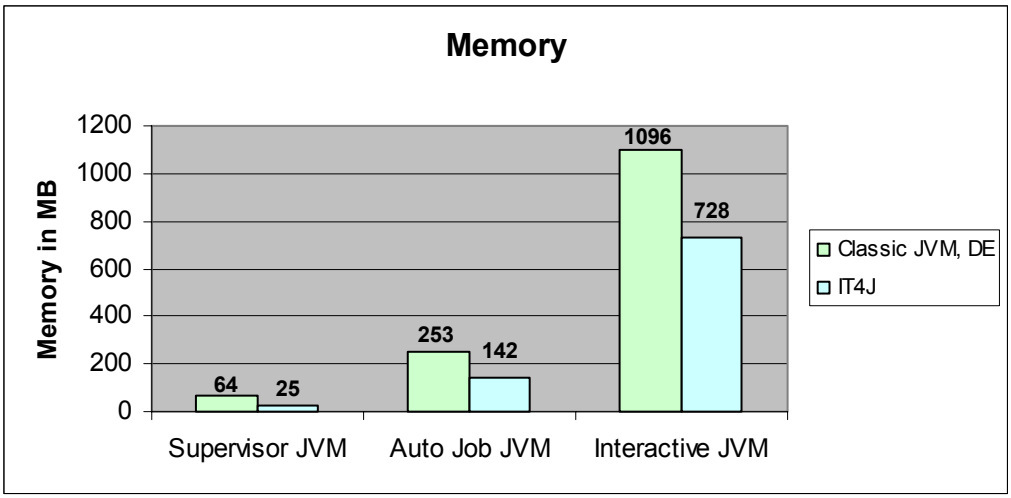
Response time. Table 9 shows the average response time for each Web browser transaction on M3 System Foundation 7.1 with IT4J. Response times do not include the average one-second think time for each transaction and they measure the entire length of the run, not just the 90 minute measurement interval.

Step	Number per Order	Average Response Time (seconds)		90th Percentile Response Time (seconds)	
		Run 1	Run 2	Run 1	Run 2
Create Order Head	1	0.132	0.132	0.17	0.17
Insert Order Line	5	0.076	0.075	0.109	0.109
Close Order	1	0.065	0.065	0.099	0.099

Table 9. Average response times for M3 System Foundation 7.1 with IT4J

Memory

As previous sections have mentioned, one key advantage of a 32-bit JVM like IT4J is reduced memory footprint. On average a 40% reduction in memory can be seen when using the 32-bit IT4J virtual machine when compare to the 64-bit Classic JVM. The tests below show that a gain of up to 40% or more is seen on the M3 System Foundation 7.1 as well.

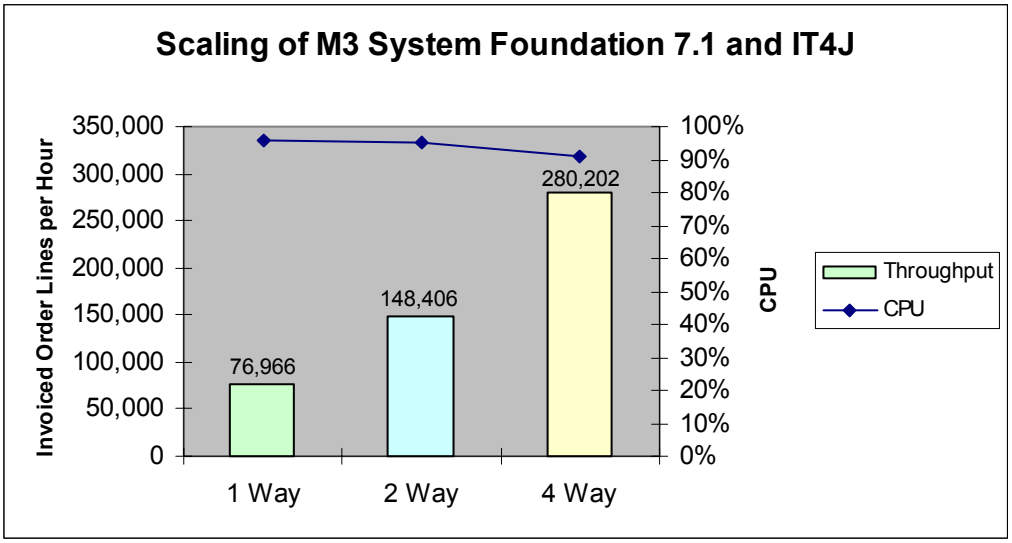


Note: Memory differences between Classic JVM with JIT and IT4J was not shown here, but there is no significant memory footprint difference between Classic DE and Classic JIT. Thus IT4J would see a similar significantly reduced memory footprint when compared to Classic JVM with JIT.

CPU Utilization
The CPU(s) experienced 90 to 100% utilization for all of the tests in this section.

Scaling Results

The scalability tests showed excellent scalability as processors were added to this configuration, demonstrating the ability of both the M3 System Foundation 7.1 and IT4J to grow along with your business. Throughput increased dramatically with each additional processor, and response time remained excellent for each of the configurations tested.



1-way
The 1-way invoiced 76,475 order lines per hour in one run and 77,456 order lines per hour in the second run. In both runs, 32 virtual users were used.

Response time. Table 10 shows the average response time for each Web browser transaction on the 1-way. Response times do not include the average one-second think time for each transaction and they measure the entire length of the run, not just the 90 minute measurement interval.

Step	Number per Order	Average Response Time (seconds)		90th Percentile Response Time (seconds)	
		Run 1	Run 2	Run 1	Run 2
Create Order Head	1	0.138	0.139	0.19	0.19
Insert Order Line	5	0.079	0.08	0.109	0.109
Close Order	1	0.067	0.066	0.109	0.109

Table 10. Average response times for the 1-way

2-way
The 2-way invoiced 148,360 order lines per hour in one run and 148,451 order lines per hour in the second run. In both runs, 63 virtual users were used.

Response time. Table 11 shows the average response time for each Web browser transaction on the 2-way. Response times do not include the average one-second think time for each transaction and they measure the entire length of the run, not just the 90 minute measurement interval.

Step	Number per Order	Average Response Time (seconds)		90th Percentile Response Time (seconds)	
		Run 1	Run 2	Run 1	Run 2
Create Order Head	1	0.152	0.152	0.21	0.21
Insert Order Line	5	0.086	0.086	0.129	0.129
Close Order	1	0.077	0.077	0.119	0.109

Table 11. Average response times for the 2-way

4-way
The 4-way invoiced 280,183 order lines per hour in one run and 280,221 order lines per hour in the second run. In both runs, 118 virtual users were used.

Response time. Table 12 shows the average response time for each Web browser transaction on the 4-way. Response times do not include the average one-second think time for each transaction and they measure the entire length of the run, not just the 90 minute measurement interval.

Step	Number per Order	Average Response Time (seconds)		90th Percentile Response Time (seconds)	
		Run 1	Run 2	Run 1	Run 2
Create Order Head	1	0.132	0.132	0.17	0.17
Insert Order Line	5	0.076	0.075	0.109	0.109
Close Order	1	0.065	0.065	0.099	0.099

Table 12. Average response times for the 4-way

CPU Utilization
The CPU(s) experienced 90 to 100% utilization for all of the tests in this section.

Summary

As the results in this paper have shown a typical customer moving to M3 System Foundation 7.1 and using IT4J can seen significantly improved runtime performance and up to a 40% or more reduction in memory requirements. Also M3 System Foundation 7.1 shows excellent scalability using IT4J on System i as processors are added to the configuration.

All of this also demonstrates the unique capabilities of the System i5 to provide superior scaling, the ability to run multiple complex workloads even on a small single processor system, the integration strength of i5/OS, DB2® UDB for i5/OS, Java, and WebSphere Application Server, and the advantages new technology such as IT4J can bring to an ISV solution. This also demonstrates the value of consolidation and integration of the Lawson M3's ISV solutions to a "One Box" solution providing customer value in the small and medium business market. Thus a single System i5 can be used for both back-end and complementary front-end applications to replace a group of heterogeneous servers, with significant price/performance and Total Cost of Ownership (TCO) benefits.

Trademarks and Disclaimers

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

Trademarks of International Business Machines Corporation in the United States, other countries, or both can be found on the World Wide Web at http://www.ibm.com/legal/copytrade.shtml.

Intel, Intel logo, Intel Inside, Intel Inside logo, Intel Centrino, Intel Centrino logo, Celeron, Intel Xeon, Intel SpeedStep, Itanium, and Pentium are trademarks or registered
trademarks of Intel Corporation or its subsidiaries in the United States and other countries.
Linux is a registered trademark of Linus Torvalds 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.
UNIX is a registered trademark of The Open Group in the United States and other countries.
Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. 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.

The customer examples described are presented as illustrations of how those customers have used IBM products and the results they may have achieved. Actual environmental costs and performance characteristics may vary by customer.

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.

All statements regarding IBM future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only.

Some information addresses anticipated future capabilities. Such information is not intended as a definitive statement of a commitment to specific levels of performance, function or delivery schedules with respect to any future products. Such commitments are only made in IBM product announcements. The information is presented here to communicate IBM's current investment and development activities as a good faith effort to help with our customers' future planning.

Performance is based on measurements and projections using standard IBM benchmarks in a controlled environment. The actual throughput or performance that any user will experience will vary depending upon considerations such as the amount of multiprogramming in the user's job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve throughput or performance improvements equivalent to the ratios stated here.

Photographs shown may be engineering prototypes. Changes may be incorporated in production models.