Redbooks
ibm.com/redbooks

IBM CICS Performance Series

A CPU Utilization Study of Java EE applications running in CICS TS V5.3

Graham Rawson

z Systems

IBM

Redpaper

Find and read thousands of IBM Redbooks publications

- Search, bookmark, save and organize favorites
- Get personalized notifications of new content
- Link to the latest Redbooks blogs and videos

Get the latest version of the Redbooks Mobile App

iOS **Download Now** Android

Promote your business in an IBM Redbooks publication

Place a Sponsorship Promotion in an IBM® Redbooks® publication, featuring your business or solution with a link to your web site.

Qualified IBM Business Partners may place a full page promotion in the most popular Redbooks publications. Imagine the power of being seen by users who download millions of Redbooks publications each year!

ibm.com/Redbooks
About Redbooks → Business Partner Programs

THIS PAGE INTENTIONALLY LEFT BLANK

Introduction

This IBM® Redpaper™ publication is a study of the performance of a Java Platform, Enterprise Edition (Java EE) application that is hosted by an IBM CICS® Transaction Server for z/OS® (CICS) V5.3 Liberty JVM server that uses the standard mode of operation that is provided by APAR PI58375. In this mode, nearly all of the application's central processing unit (CPU) consumption can be off-loaded to IBM System z® Integrated Information Processors (zIIPs), which minimizes the total cost of computing for such a workload.

The target audience for this Redpaper publication is Capacity Planners and Performance Analysts who might need to understand how to configure an IBM CICS Liberty JVM server to run Java EE applications with minimal general processor use.

This Redpaper publication is one in a series that is focused on CICS performance. It is written by a member of the IBM Hursley CICS development community. The subject matter in this series is based on feedback from CICS customers.

Disclaimer: All performance data that is contained in this Redpaper publication was obtained in the specified operating environment and configurations and must be considered as an example. Performance characteristics of other operating environments might differ.

IBM does not represent, warrant, or guarantee that the same or similar results will be achieved in a user's environment as those results that are reported in this Redpaper publication.

Overview

This Redpaper publication contains a description of the Java EE application that was used in this performance study. It also includes a description of the environment in which the performance measurements were made. The results from the performance measurements and analysis of these results also are presented. This Redpaper publication includes the following topics:

- A description of the Java EE application.
- A description of the hardware and software that comprises the measurement environment.
- An overview of standard and integrated modes of operation of the Liberty JVM server that is provided by CICS TS V5.3.

- A comparison of the z/OS processor resources that are used by running the Java EE application in the two modes of operation of the Liberty JVM server that is provided by CICS TS V5.3.
- A comparison of the z/OS processor resources that are used by co-locating the Java EE application and database server in a single z/OS LPAR or in two LPARs with the CICS TS server that is separated from the IBM DB2® database server.
- An explanation of the performance metrics that were used in this study.
- An analysis of the results in terms of CPU cost, zIIP usage, and eligibility and request response time.

Executive summary

The DayTrader 3 Java EE benchmarking application (as featured in numerous IBM WebSphere® Application Server studies) was used to study the reduction in general processor usage that can be achieved by using CICS Liberty JVM server in standard mode.

The following configurations were studied:

- CICS Liberty JVM server in standard mode on LPAR 1 and IBM DB2 database server on LPAR 2
- CICS Liberty JVM server in standard mode and DB2 database server both on LPAR 1
- CICS Liberty JVM server in integrated mode on LPAR 1 and DB2 database server on LPAR 2

The mean processor costs and response times per web request are shown in Figure 1 for the configurations that were studied at a rate of approximately 17,500 requests per second. CP represents general processor time and zIIP represents zIIP time that was used on an LPAR.

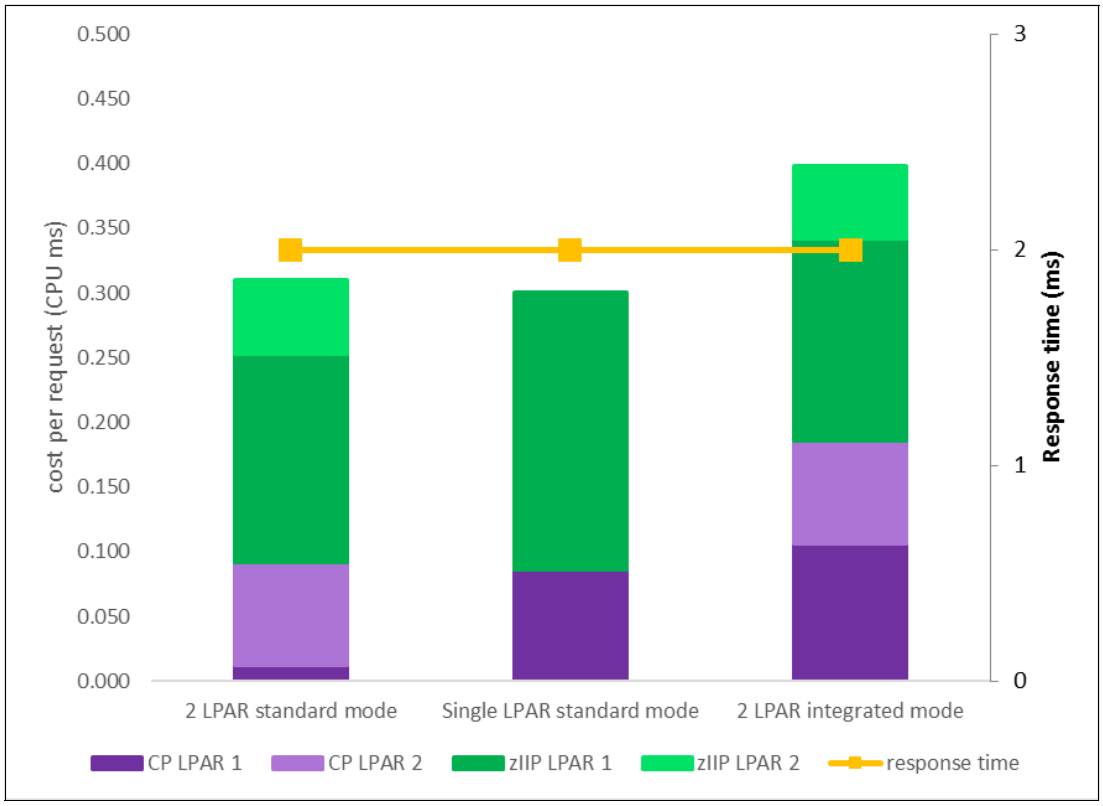

Figure 1 Processor usage and response times of the studied configurations

Minimizing general processor use is a performance objective for many CICS customers because this approach also can reduce software costs.

When a Java EE application is hosted in a CICS Liberty JVM server in standard mode, nearly all of the processing uses zIIP processors; therefore, the general processor time that is used by CICS is low. This processing shown in Figure 1 on page 3 in the “2 LPAR standard mode” case where “CP LPAR 1” is low.

Consolidating CICS and the DB2 database server onto a single LPAR realizes some small savings in total general processor usage through saving the cost of LPAR intercommunication; in this case, by using TCP/IP over a Shared Memory Communications – Direct-Access Method (SMC-D) network. This approach is shown in Figure 1 on page 3 in which the sum of CP LPAR 1 and CP LPAR 2 that is shown for the 2 LPAR standard mode configuration is compared with CP LPAR 1 that is shown for the Single LPAR standard mode configuration.

The use of CICS Liberty in integrated mode increases overall processor cost when all the threads in the CICS JVM Liberty server are CICS enabled. Much of this increase is not zIIP-eligible and runs on general processors. This approach is shown in Figure 1 on page 3 by comparing the CP LPAR 1 data in 2 LPAR standard mode with 2 LPAR-integrated mode.

If the Java EE application has no need for CICS functions, the use of a CICS standard mode Liberty server can reduce general processor usage.

Average response times do not differ significantly for the configurations that were studied.

Standard mode Liberty in CICS

Before the introduction of CICS standard mode Liberty, applications that were hosted by a CICS Liberty JVM server ran on threads that were CICS enabled. *CICS enabled* means that applications are integrated with CICS security, CICS unit of work (UOW) management, and can use the Java class library for CICS (JCICS). This integration of CICS functions incurs some extra processing costs, much of which is not eligible for offload to a zIIP processor.

Standard mode was introduced as an alternative to this default-integrated mode so that applications can be run by a CICS Liberty JVM server to take advantage of Liberty services, management, and security but are not fully integrated with CICS. In this mode, threads are not CICS enabled by default, so they do not run within a CICS transaction context and do not have access to CICS functions.

A Java EE application that is hosted in standard mode can be coded to run under a CICS enabled thread by submitting work to the `CICSExecutorService` by using the `runAsCICS()` method, which gives the application full access to CICS resources by using the JCICS API. The use of this technique ensures that the overhead of CICS enablement is suffered only by threads that specifically need this capability.

Standard mode provides a performance benefit for hosting Java EE applications that have no requirement to access CICS resources and can be ported and deployed from other platforms to CICS without application change. DayTrader 3 is an example of such an application, which is described next.

DayTrader 3 application

The Java EE application that was chosen for this study is the DayTrader 3 benchmark sample. For more information about how to download this sample application, see “Other resources for more information” on page 15.

DayTrader 3 is a Java EE 6 application that provides an online stock trading system. It uses Java EE Web Profile and Full Profile technologies, such as servlets, Java Server Pages (JSP), JavaServer Faces (JSF), Enterprise java Beans (EJBs) Java database connectivity (JDBC), Java Message Service (JMS), Message-Driven Beans (MDB), and Web Services that use JAX-RS.

DayTrader 3 was used in various WebSphere® Application Server performance studies, such as the session from the IBM InterConnect 2015 conference. For more information, see “Other resources for more information” on page 15. DayTrader 3 is an ideal candidate for similar performance studies that use the CICS Liberty JVM server.

Note: DayTrader 3 does not use CICS Java API (JCICS) to access any CICS resources.

The data that was used in this study consists of the default 15,000 trader accounts and 10,000 company share quotations that are stored in a DB2 for z/OS database system as 6 tables and 11 indexes.

The DayTrader 3 application is driven by requests that are entered from a web browser, as shown in Figure 2 on page 6. The DayTrader 3 download package includes workload scripts that can be run by Apache JMeter or IBM WebSphere Studio Workload Simulator. The supplied `daytrader3.jmx` script comprises a mix of DayTrader 3 request types and the following default mix was used in this study:

- 85% Servlet/JSP requests
- 10% JSF requests
- 5% JAX-RS requests

The workload simulator can be configured to simulate many clients. For this study, 50 clients were used with no delay time.

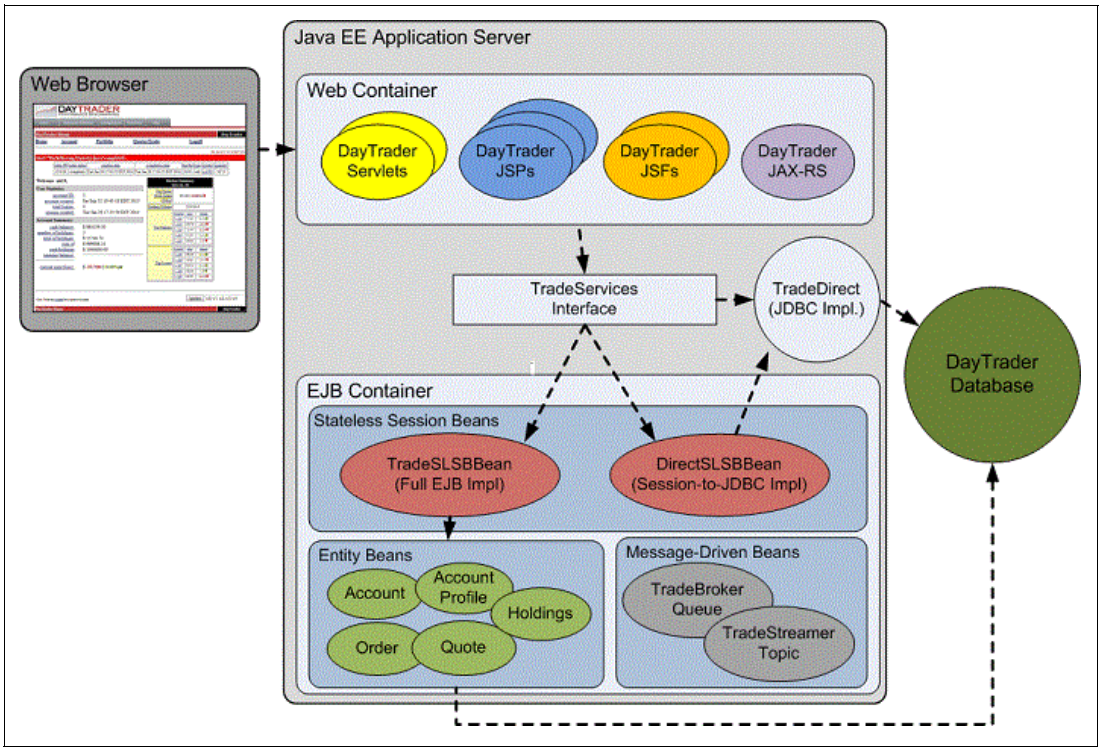

Figure 2 Overview of the DayTrader 3 application architecture

Nested transaction scope within the DayTrader 3 application

DayTrader 3 uses nested JTA UserTransactions.

CICS Liberty in integrated mode supports JTA UserTransactions. However, nested JTA UserTransactions are not supported in integrated mode because CICS does not support nested units-of-work (UOW).

To run DayTrader 3 successfully in CICS integrated mode, Liberty JTA transaction support was disabled by adding the following `server.xml` configuration element:

```
<cicsts_jta integration="false"/>
```

With this setting, CICS treats the entire web request as a single UOW.

Nested JTA transactions are supported by CICS standard mode Liberty. In this mode, Liberty does not coordinate with CICS recovery manager and so does not need to conform to the CICS UOW restriction.

Test environment and schematics

This section describes the hardware and software configurations that were used to run the performance measurements.

The software configurations that were used for this study are shown in Figure 3 and Figure 4.

Figure 3 shows CICS Liberty and DB2 database server on different LPARs. In this case, the network link between the CICS and DB2 system connects different TCP/IP hosts and accesses a remote database by using a type 4 JDBC driver connection.

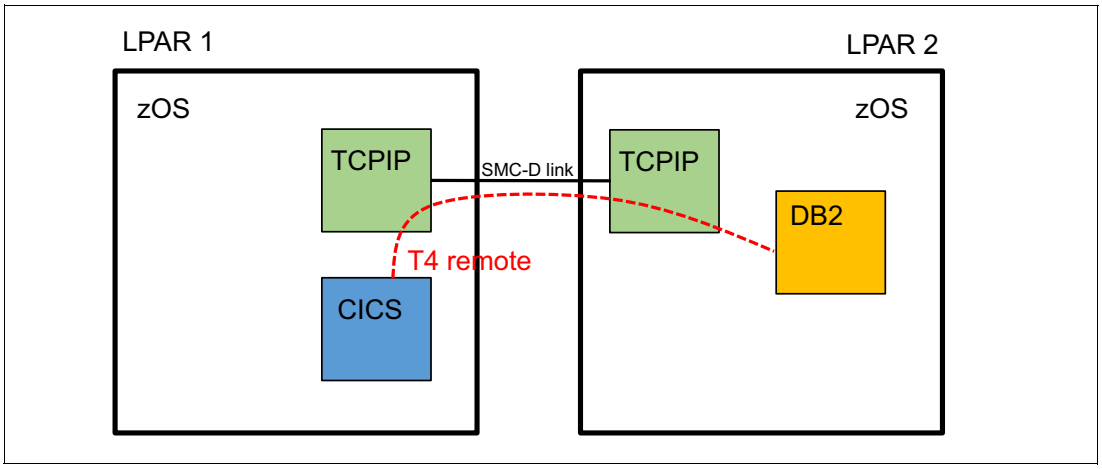

Figure 3 Two LPAR configuration

Figure 4 shows CICS Liberty and DB2 co-located on the same LPAR. In this case, the network link is within the same TCP/IP host and uses a type 4 JDBC driver connection by using fast local sockets to an IP address within the same TCP/IP stack.

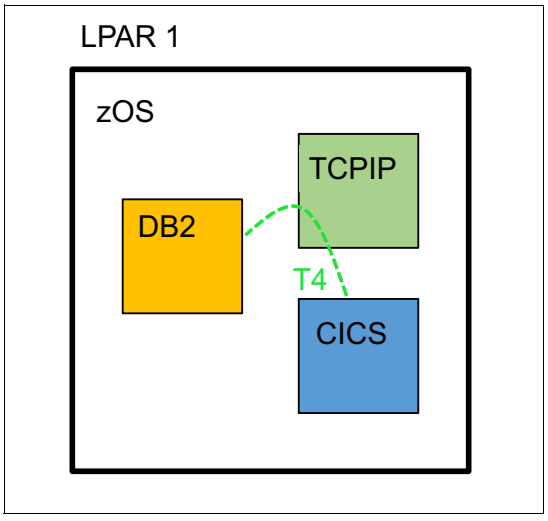

Figure 4 Single LPAR configuration

Hardware

The measurement environment comprises LPARs of the same IBM z13™ system. Each LPAR features dedicated processors and memory. The TCP/IP services on each LPAR are connected by an SMC-D link. This configuration is a connectivity protocol for IBM z13 systems that provides superior data transfer rates and lower CPU costs compared to alternatives, such as IBM HiperSockets™ or an Open Systems Adapter (OSA). For more information, see “Other resources for more information” on page 15.

The following system configurations were used:

- z/OS LPAR 1:
  - A z13 system 2964-7B3 LPAR (equivalent to a z13 system 2964-703)
  - Three dedicated general-purpose Central Processors (CP)
  - Eight dedicated zIIPs in SMT=2 mode
  - 16 GB real storage
- z/OS LPAR 2:
  - A z13 system 2964-7B3 LPAR (equivalent to a z13 system 2964-704)
  - Four dedicated CPs
  - Two dedicated zIIPs in SMT=2 mode
  - 16 GB real storage

Software

Both systems in this study use the following software:

- IBM z/OS V2R2
- CICS V5.3 including APAR PI58375
- Java V8.0SR2
- IBM DB2 for z/OS 11.01.0

A single CICS region is used in this study.

A DB2 for z/OS database server is used as the data source for the JDBC requests. Local and remote database servers are used for this study. The local database server runs in the same LPAR as the CICS system. The remote database server runs on another LPAR in the same z/OS sysplex, which is connected by using an SMC-D connection between TCP/IP services.

Performance measurement procedure

The same procedure was carried out for each measurement and consisted of the following steps:

1. The CICS system was COLD started.
2. The DB2 tables and indexes were recreated.
3. The DB2 tables and indexes were populated with 15,000 trader accounts and 10,000 company share quotations by the DayTrader 3 application.

4. The DayTrader 3 simulator script that comprises 50 web clients was run for 30 minutes. The workload rapidly reached steady state throughput rate of about 17,300 web requests per second. Measurements were taken in the final 5 minutes of this 30-minute period. The data that is shown in this study is for the final minute of each 5-minute period.
5. IBM z/OS Resource Measurement Facility™ (RMF™) data was collected at 1-minute intervals.

Performance metrics

As described in “Results” on page 11, the following metrics were reported:

- Request rate
- Response time
- CP (general processor) usage
- zIIP usage

CP and zIIP utilization percentage values are extracted from RMF workload activity data. RMF reports CPU utilization that is based on uniprocessor capacity, which means that a value of 100% represents the capacity of one processor. As an example of how CP and zIIP data is derived, consider the extract of an RMF report that is shown in Figure 5 for a single CICS region.

```
                                               W O R K L O A D   A C T I V I T Y
                                                                                                                     PAGE  29
      z/OS V2R2                SYSPLEX PLEX3                START 06/11/2016-15.06.00 INTERVAL 000.04.59   MODE = GOAL
                         CONVERTED TO z/OS V2R2 RMF       END   06/11/2016-15.11.00

REPORT BY: POLICY=POLICY                         REPORT CLASS=CICS2A20
                                                 DESCRIPTION =

-TRANSACTIONS-  TRANS-TIME HHH.MM.SS.TTT  --DASD I/O--  ---SERVICE---   SERVICE TIME  ---APPL %---  --PROMOTED--  ----STORAGE----
AVG       1.00  ACTUAL                 0  SSCHRT   0.0  IOC     3056K   CPU 1321.472  CP      3.08  BLK    0.000  AVG    614117.2
MPL       1.00  EXECUTION              0  RESP     0.0  CPU   105192K   SRB    0.223  AAPCP   0.00  ENQ    0.000  TOTAL  614121.3
ENDED        0  QUEUED                 0  CONN     0.0  MSO         0   RCT    0.000  IIPCP   3.00  CRM    0.000  SHARED     0.00
END/S     0.00  R/S AFFIN              0  DISC     0.0  SRB    17780    IIT    0.000                LCK    0.392
#SWAPS       0  INELIGIBLE             0  Q+PEND   0.0  TOT   108266K   HST    0.000  AAP      N/A  SUP    0.000  -PAGE-IN RATES-
EXCTD        0  CONVERSION             0  IOSQ     0.0  /SEC  360889    AAP      N/A  IIP   278.53                SINGLE      0.0
AVG ENC   0.00  STD DEV                0                                IIP 1312.441                              BLOCK       0.0
REM ENC   0.00                                          ABSRPTN  361K                                             SHARED      0.0
MS ENC    0.00                                          TRX SERV 361K                                             HSP         0.0
```

Figure 5 RMF Workload Activity extract

For the 4 minute 59 s interval that was reported, the APPL% section shows that 3.08% of general processor time was used (represented by the CP data) and of that time that was spent on a general processor, 3.00% was eligible to use a zIIP processor (represented by the IIPCP data). zIIP eligible work is run on a general processor if all the available zIIP processors are busy.

The report also shows 278.53% zIIP time was used (represented by the IIP data). A value of 100% means that one processor is utilized, so a value of 278.53% for zIIP usage indicates that the workload is consuming the resources of nearly three zIIP processors.

RMF reports performance data for all the following subsystems that are contributing to this workload:

- CICS
- TCP/IP (for network traffic)
- DB2
- Distributed data facility (DDF), which is the distributed data facility component of DB2 that runs requests that use the type 4 JDBC driver.

Response times and request counts are extracted from the simulator tool reports, as shown in Figure 6. In this example, an HTTP request rate of 17,364 per second with an average response time of 2 ms is reported.

```
IWLO058I Time limit of 360 seconds has been reached.  Shutting down.
IWLO038I Run time = 00:06:05
IWLO007I Clients completed = 0/50
IWLO059I Page elements = 6346682
IWLO060I Page element throughput = 17363.629 /s
IWLO059I Transactions = 5461633
IWLO060I Transaction throughput = 14942.258 /s
IWLO059I Network I/O errors = 0
IWLO059I Web server errors = 0
IWLO059I Num of pages retrieved = 6346679
IWLO060I Page throughput = 17363.617 /s
IWLO060I HTTP data read = 47225.809 MB
IWLO060I HTTP data written = 2996.517 MB
IWLO060I HTTP avg. page element response time = 0.002
```

Figure 6 Simulator tool report extract

For more information about performance measurement procedures, see *IBM CICS Performance Series: Effective Monitoring for CICS Performance Benchmarks*, REDP-5170.

Configurations measured

This section describes the configurations that were used for the results that were reported in this study.

Standard mode on two LPARs

In this configuration, the CICS Liberty server that is running in standard mode runs on LPAR 1. The JVM profile for the JVM server in this case includes the following option:

`CICS_WLP_MODE=STANDARD`

The DB2 database server runs on LPAR 2. The dataSource definitions in `server.xml` use a type 4 JDBC connection to the remote database system parsing a `serverName` parameter that in this case specifies an SMC-D connection (see Example 1).

Example 1 Running on LPAR 2

```
<dataSource connectionManagerRef="conMgr1" id="TradeDataSource"
isolationLevel="TRANSACTION_READ_COMMITTED" jdbcDriverRef="DB2JCC"
jndiName="jdbc/TradeDataSource" statementCacheSize="60"> <properties
databaseName="DSNV11P3" driverType="4" password="********" portNumber="41100"
serverName="10.20.4.15" user="WSADMIN"/> </dataSource>
```

The use of SMC-D can be verified by using the z/OS operator command

`D TCPIP,TCPIP2,NETSTAT,ALL`, which reports SMC status information (see Example 2 on page 11).

Example 2 SMC status

```
SMC INFORMATION:
  SMCDSTATUS:       ACTIVE
    LOCALSMCLINKID: 21020002         REMOTESMCLINKID:    23020002
    LOCALSMCRCVBUF: 64 K              REMOTESMCRCVBUF:    64 K
```

Standard mode on a single LPAR

In this configuration, the CICS Liberty JVM server is running in standard mode and the DB2 database server runs on the same LPAR. The dataSource definitions in `server.xml` use a type 4 JDBC connection to the database system that specifies a serverName IP address of 127.0.0.1, which is the local TCP/IP system.

Integrated mode on 2 LPARs

This configuration differs from the “2 LPAR Standard” configuration only in that the CICS Liberty JVM server is defined to run in integrated mode. The JVM profile for the JVM server in this case uses the default setting of INTEGRATED for the CICS_WLP_MODE option.

Results

This section describes the performance measurements of the configurations that were studied, reporting request rates, CP and zIIP usage, and average response times.

Standard mode on 2 LPARs

Table 1 lists the request rate, response time, and processor usage for DayTrader 3 that was hosted by a CICS Liberty JVM server that is running in standard mode and a database server on a second LPAR. The following request rate and average response times were observed:

- Request rate: 17,364 requests per second
- Average response time: 2 ms

Table 1 CPU usage for DayTrader 3 running on 2 LPARs in standard mode

	CP%	**zIIP%**	**Total%**
CICS only	3.08	278.53	281.61
All LPAR 1	20.71	278.53	299.24
All LPAR 2	139.27	99.14	238.41
Total for both LPARs	159.98	377.67	537.65

The same data that is expressed as cost per request is listed in Table 2.

Table 2 CPU usage per request by DayTrader 3 running on 2 LPARs in standard mode

	CP (ms)	zIIP (ms)	Total (ms)
CICS only	0.002	0.160	0.162
All LPAR 1	0.012	0.160	0.172
All LPAR 2	0.080	0.057	0.137
Total for both LPARs	0.092	0.218	0.310

Observation: The CICS Liberty JVM server in standard mode uses little general processor time to run DayTrader 3. The CICS only data in Table 2 shows that, of the 0.162 ms of total CPU time to run a request, approximately 99% (0.160 ms) was run on zIIP processors.

A requirement for general processor time by other software components that are running on the system under test, such as TCP/IP services, adds to the total general processor use for running the workload.

Standard mode on a single LPAR

Table 3 lists the request rate, response time, and processor usage for DayTrader 3 that is hosted by a CICS Liberty JVM server that is running in standard mode and a database server on the same LPAR. The following request rate and average response times were observed:

- Request rate: 18,636 requests per second
- Average response time: 2 ms

Table 3 CPU usage per request by DayTrader 3 running on single LPARs in standard mode

	CP%	zIIP%	Total (ms)
CICS only	4.28	295.12	299.40
All LPAR 1	160.99	398.12	559.11

The same data that is expressed as cost per request is listed in Table 4.

Table 4 CPU usage per request by DayTrader 3 running on a single LPAR in standard mode

	CP (ms)	zIIP (ms)	Total (ms)
CICS only	0.002	0.158	0.161
All LPAR 1	0.086	0.214	0.300

Observations: Running the CICS Liberty JVM server in standard mode and DB2 database server on the same LPAR saves the overhead of communication between LPARs. This reduction is shown as a small decrease in total processor cost per request, mainly in CP usage (from 0.092 ms to 0.086 ms). This reduction occurs because data does not need to be transferred between LPARs by using SMC-D. There also is a small increase in the request rate compared to the standard mode on 2 LPARs case.

Integrated mode on 2 LPARs

Table 5 lists the request rate, response time, and processor usage for DayTrader 3 that is hosted by a CICS Liberty JVM server that is running in integrated mode on LPAR 1 and a DB2 database server on LPAR 2. The following request rate and average response times were observed:

- Request rate: 17,395 requests per second
- Average response time: 2 ms

Table 5 CPU usage per request by DayTrader 3 running on 2 LPARs in integrated mode

	CP%	**zIIP%**	**Total (ms)**
CICS only	164.89	271.01	435.90
All LPAR 1	184.09	271.01	455.10
All LPAR 2	138.82	97.93	236.75
Total for both LPARs	322.91	368.94	691.85

The same data that is expressed as cost per request is listed in Table 6.

Table 6 CPU usage per request by DayTrader 3 running on 2 LPARs in integrated mode

	CP (ms)	**zIIP (ms)**	**Total (ms)**
CICS only	0.095	0.156	0.251
All LPAR 1	0.106	0.156	0.251
All LPAR 2	0.080	0.056	0.136
Total for both LPARs	0.186	0.212	0.398

Observations: Threads that are running in CICS Liberty in integrated mode are fully CICS enabled. They are dispatched on CICS T8 TCBs and are integrated with CICS functions, but this integration comes with some extra costs. Profiling analysis of this workload showed that most of this extra cost was caused by CICS task management and more UNIX pthread and Java thread management.

Comparing the CPU usage per request results for standard mode and integrated mode, it can be seen that general processor time is increased from 0.092 ms to 0.186 ms but zIIP usage is barely affected.

Conclusions

The use of CICS Liberty JVM server in standard mode enables most of the processing of a Java EE application that does not need access to CICS resources to be run almost exclusively on zIIP processors. DayTrader 3 is an example of such an application and this Redpaper publication described how little general processor time is needed to run this application, even at relatively high throughput rates of over 17,000 web requests per second.

Considering the total general processor usage for both LPARs that were used in this study, consolidating CICS and the DB2 database server onto a single LPAR reduces general processor usage. This reduction is achieved from saving the cost of TCP/IP intercommunication between LPARs, in this case by using SMC-D.

The use of CICS Liberty in integrated mode, when all the threads in the CICS JVM Liberty server are CICS enabled, increases overall processor cost. In this mode, each application thread in the CICS Liberty server creates a CICS transaction and is dispatched on a CICS managed TCB. Much of this extra thread management is not zIIP eligible and runs on general processors.

If the Java EE application has no need for CICS functions, the use of standard mode CICS Liberty server can save a significant amount of general processor usage. Running in this mode means that no CICS task is created to run the CICS Liberty thread and functions, such as CICS security or access to CICS resources that use the JCICS API, are not available. Because no CICS task is created to run in this mode, no CICS monitoring data or transaction-related statistics is created.

Average response times at the simulated web clients do not differ significantly for the configurations that were studied.

Other resources for more information

For more information, see the following resources:

- *IBM CICS Performance Series: Effective Monitoring for CICS Performance Benchmarks*, REDP-5170:

  http://www.redbooks.ibm.com/abstracts/redp5170.html

- DayTrader 3:

  https://ibm.biz/BdrHHm

- DayTrader 3 is used in many performance evaluations of WebSphere Application Server Liberty Profile:

  https://ibm.biz/BdrHHn

- Shared Memory Communications - Direct Memory Access (SMC-D) provides high-speed communications between z Systems servers:

  http://www.ibm.com/common/ssi/cgi-bin/ssialias?htmlfid=ZSB03082USEN

Author

This Redpaper publication was produced by a team of specialists from around the world working at the International Technical Support Organization, Raleigh Center.

Graham Rawson is a CICS Performance Analyst within the CICS Development group that is based at the IBM Laboratory in Hursley, England. Graham has worked for IBM for 30 years in various roles supporting CICS Transaction Server and IBM Java technology. Graham holds a B.Sc. (Hons) in Chemistry from the University of East Anglia (Norwich, England) and a Certificate in Software Engineering from the University of Oxford (England).

Thanks to the following people for their contributions to this project:

Michael Casile
IBM Competitive Project Office for his invaluable assistance in preparing this Redpaper publication.

Hernan Cunico, LindaMay Patterson
International Technical Support Organization, Raleigh Center

Now you can become a published author, too!

Here's an opportunity to spotlight your skills, grow your career, and become a published author—all at the same time! Join an ITSO residency project and help write a book in your area of expertise, while honing your experience using leading-edge technologies. Your efforts will help to increase product acceptance and customer satisfaction, as you expand your network of technical contacts and relationships. Residencies run from two to six weeks in length, and you can participate either in person or as a remote resident working from your home base.

Find out more about the residency program, browse the residency index, and apply online at:

ibm.com/redbooks/residencies.html

Stay connected to IBM Redbooks

- Find us on Facebook:

  http://www.facebook.com/IBMRedbooks
- Follow us on Twitter:

  http://twitter.com/ibmredbooks
- Look for us on LinkedIn:

  http://www.linkedin.com/groups?home=&gid=2130806
- Explore new Redbooks publications, residencies, and workshops with the IBM Redbooks weekly newsletter:

  https://www.redbooks.ibm.com/Redbooks.nsf/subscribe?OpenForm
- Stay current on recent Redbooks publications with RSS Feeds:

  http://www.redbooks.ibm.com/rss.html

Notices

This information was developed for products and services offered in the US. This material might be available from IBM in other languages. However, you may be required to own a copy of the product or product version in that language in order to access it.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:
IBM Director of Licensing, IBM Corporation, North Castle Drive, MD-NC119, Armonk, NY 10504-1785, US

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION “AS IS” WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some jurisdictions do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM websites are provided for convenience only and do not in any manner serve as an endorsement of those websites. The materials at those websites are not part of the materials for this IBM product and use of those websites is at your own risk.

IBM may use or distribute any of the information you provide in any way it believes appropriate without incurring any obligation to you.

The performance data and client examples cited are presented for illustrative purposes only. Actual performance results may vary depending on specific configurations and operating conditions.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

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

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to actual people or business enterprises is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. The sample programs are provided “AS IS”, without warranty of any kind. IBM shall not be liable for any damages arising out of your use of the sample programs.

Trademarks

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corporation, registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the web at “Copyright and trademark information” at http://www.ibm.com/legal/copytrade.shtml

The following terms are trademarks or registered trademarks of International Business Machines Corporation, and might also be trademarks or registered trademarks in other countries.

CICS®	Redbooks®	System z®
DB2®	Redpaper™	WebSphere®
HiperSockets™	Redbooks (logo)  ®	z/OS®
IBM®	Resource Measurement Facility™	z13™
IBM z13™	RMF™	

The following terms are trademarks of other companies:

Java, and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

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

REDP-5361-00

ISBN 0738455474

Printed in U.S.A.