IMS Performance Evaluation on IBM z16

IMS Performance Evaluation Team (IBM Silicon Valley Laboratory)
November 2022

Contributors:
Justin Soe, Osamu Nakayama, Kly Florence, Kevin Hite, and Bruce Naylor

Abstract

This paper presents the performance characteristics of IBM® z16™ (z16), the latest member of the IBM Z® platform launched in April 2022. The IMS Performance Evaluation Team at IBM Silicon Valley Laboratory evaluated z16 using Information Management System (IMS™) 15.3 and compared it with the prior IBM Z platform, IBM z15® (z15). These evaluations used different IMS workload types under similar software configurations.

The z16 is IBM's next-generation system built with the IBM Telum™ processor continues to deliver the best-of-breed security, resiliency, and scalability relied on for mission-critical workloads and as an essential element of hybrid cloud infrastructures. The IBM z16 features an integrated on-chip Artificial Intelligence (AI) accelerator to deliver speed and scale to infuse real-time transaction analysis, a quantum-safe technology to help protect against cyber threats, and a flexible infrastructure to meet the constant changes in business needs.

The z16 and IMS 15.3 offer a great combination of performance and security features for IT systems to keep up with growing business needs while keeping critical data safe and secure from internal and external threats. IMS applications may benefit from many new features including the flexibility of capacity to dynamically shift system resources across locations to proactively avoid disruptions, modernization of applications and data in a hybrid cloud environment, and reduction of cost to keep up with ever changing regulations.

IBM Z
San Jose, CA

Note: Before you use this information and the product it supports, read the information in "Notices" section.

Note: Performance is based on measurements and projections using IMS benchmarks in a controlled environment. The results that any user will experience will vary depending upon many factors, including 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, results may vary significantly, and no assurance can be given that an individual user will achieve results like those stated here. Results should be used for reference purposes only.

The test scenarios (hardware configuration and workloads) used in this document to generate performance data are not considered 'best performance case' scenarios. Performance may be better or worse depending on the hardware configuration, data set types and sizes, and the overall workload on the system.

The information contained in this document has not been submitted to any formal IBM test and is distributed on an "AS IS" basis without any warranty either expressed or implied. The use of this information or the implementation of any of these techniques is a customer responsibility and depends on the customer's ability to evaluate and integrate them into their operational environment. While each item may have been reviewed by IBM for accuracy in a specific situation, there is no guarantee that the same or similar results will be obtained elsewhere. Customers attempting to adapt these techniques to their own environments do so at their own risk.

The information provided in this paper was obtained at the IBM Silicon Valley Laboratory and is intended for migration and capacity planning purposes.

Table of Contents

1 Introduction

The z16 is IBM's next generation system that can offer high performance, resiliency, scalability, and security for today's business. The z16 server is the first system built with the *IBM Telum processor* which is designed to help businesses:

- Optimize decision making with the on-chip Accelerator for AI, delivering high-speed and scale for workloads with no impact on service.
- Leverage the industry-first quantum-safe system protection for today's data from current and future threats from quantum computers with quantum-safe APIs.
- Enhance resiliency with flexible capacity to remotely shift capacity and production workloads between z16 systems at different locations avoiding disruptions from impending weather disasters, wildfire, or planned outages.
- Modernize and integrate mission-critical applications and data in a hybrid cloud environment with industry-standard tools like Red Hat® OpenShift® and IBM Cloud Paks®.

The Telum processor improves system performance and scalability with 1.5x more cache capability per core over the IMS z15. The new processor design has a new cache hierarchy, on-chip AI accelerator shared by the processor unit (PU) cores, transparent memory encryption, and increased uniprocessor capability. The on-chip AI scoring logic provides sub-microsecond AI inferencing for deep learning and complex neural network models.

The z16 enhancements in resiliency allow remote shift in capacity and production workloads between the z16 systems at different sites without any on-site personnel intervention. This capability is designed to proactively avoid disruptions from unplanned events as well as from outages due to planned maintenance.

Enterprise customers may be constantly challenged with data breaches. Enabling encryption through application code is not a simple task and requires a huge investment. Customers are required to encrypt client data through data security and regulatory compliance such as the General Data Protection Regulations (GDPR) policy in Europe and the Health Insurance Portability and Accountability Act of 1996 (HIPAA). The z16, along with pervasive encryption introduced in IBM z14®, will be able to protect data from today's cyber threats and future quantum attacks through the quantum-safe system. The quantum-safe technology with built-in dual signature scheme will protect the z16 firmware from quantum attacks with no configuration enablement.

2 Executive Overview

The z16 performance evaluations using the IMS 15.3 base workloads demonstrate that the latest release of the IBM zSystems family of mainframes has improved CPU efficiency and performance (depending on the environment and configuration). While performance in specific production environments will vary, results of IBM internal testing in a controlled laboratory environment revealed that the z16 can perform as described below:

- An improvement of up to 20% in ITR (Internal Transaction Rate) on the IBM z16 depending on the performance workload.
- The service time per transaction in the different components has significant improved responses.
- The total IMS response time shows either flat performance or an improvement on the IBM z16.

3 Performance Environment Information

This section explains the performance evaluation environment at Silicon Valley Lab including hardware and software specifications.

3.1 Hardware Environment

All measurements were conducted on the z16 Model 3931 (A01) and z15 Model 8561 (T01) as shown in Figure 1 and Figure 2, respectively.

3.1.1 z16 Processor

The z16 processor is the newest member of the IBM zSystems family that ensures continuity and upgradeability from the z15.

The main features of the z16 technology include:

- IBM Telum processor chip
- Dedicated on-chip accelerator for AI inference
- 7nm Extreme ultraviolet (EUV) silicon lithography technology
- 5.2 GHz system frequency
- 10 TB of main memory per drawer
- 4-drawer 200 cores system max configuration
- 256 GB Hardware System Area (HSA)
- Levels of cache:
  - First-level cache (L1): 256 KB per PU core
  - Second-level cache (L2): 32 MB semi-private per PU core
  - Third-level cache (L3): 256 MB virtual per chip
  - Fourth-level cache (L4): 2 GB virtual per drawer

Figure 1: IBM z16 Model 3931 (A01)

3.1.2 z15 Processor

The z15 processor is the prior IBM zSystems family processor with improved processing power, flexible configuration, and fit-for-purpose systems.

The main features of the z15 technology include:

- 14nm FINFET SOI technology
- 5.2 GHz system frequency
- 12 core per processor chip
- 4-processor units + 1-SC chip per drawer
- 190 cores system max configuration
- 256 GB Hardware System Area (HSA)
- Levels of cache:
  - First-level cache (L1): 128 KB per PU core
  - Second-level cache (L2): 48 MB per PU core
  - Third-level cache (L3): 256 MB shared per chip
  - Fourth-level cache (L4): 960 MB virtual per drawer

Figure 2: IBM z15 Model 8561 (T01)

3.1.3 Storage

IBM System Storage® DS8000® series latest model DS8900F were used with 64 real volumes and 128 alias volumes per LCU using dynamic Hyper Parallel Access Volumes (HyperPAVs). The DASD volumes and paths for each measurement evaluation remained consistent for comparison purposes with the z15 and z16.

3.1.4 Coupling Facility

All measurements were performed with Internal Coupling Facilities (ICFs) with Coupling Facility Control Code (CFCC) level 24 for z15 and level 25 for z16.

3.2 Software Environment

Table 1 lists the software components and version level information.

Table 1: Software Components and Version Levels

Component	Details
IBM z/OS	z/OS® Version 2 Release 5
IMS	IMS™ Version 15 Release 3
IRLM	IRLM Version 2.3
CICS	CICS® Transaction Server for z/OS®, Version 5 Release 6 connecting to IMS Database Manager (DB)
Workload Driver	Java Asynchronous Socket Channel
Java™	Java 8 Service Release 7
z/OS Connect	z/OS Connect 3.0.24
Db2	Db2® Version 12

4 Workloads

IMS supports multiple types of enterprise databases and communication access methods so that customers can exploit the technology best suited for their requirements. This section describes various types of workloads used in the IMS 15.3 performance evaluations to exercise specific IMS code paths:

- Fast Path (FP)
- Full Function (FF) with High Availability Large Database (HALDB)
- Shared Message Queues (SMQ)
- Open Transaction Manager Access (OTMA)
- CICS IMS Database Control (DBCTL)
- IMS -Db2
- Java Message Processing (JMP)
- Open Database Manager (ODBM)

These workloads were used to simulate typical banking, warehouse, hotel, and inventory customer-like workloads issuing read, replace, delete, and insert calls.

4.1 Full Function (FF) with High Availability Large Database Workload

IMS supports multiple forms of enterprise databases so that various application requirements can be met by exploiting the database technology best suited for the user's requirements.

Full-function databases are accessed through Data Language Interface (DL/I) calls and can be processed by application programs running in IMS dependent regions, CICS, z/OS WebSphere® Application Server (WAS), Db2 Stored Procedures, and through the IMS Open Database Access (ODBA) interface. IMS dependent region types include:

- IMS Fast Path Program (IFP)
- Message Processing Program (MPP)
- Batch Message Processing Region (BMP)
- Java Message Processing Region (JMP)
- Java Batch Processing Region (JBP)

Full-function databases can store data using Virtual Storage Access Method (VSAM), a native z/OS access method, or Overflow Sequential Access Method (OSAM), an IMS-specific access method that optimizes the I/O channel program for IMS access patterns.

A High Availability Large Database (HALDB) is a partitioned full-function database that allows the grouping of full-function database records into sets of partitions or replicates that are treated as a single database.

This section details the database description, application transactions and the workload transaction distribution for the full function with HALDB workload.

4.1.1 Database Description

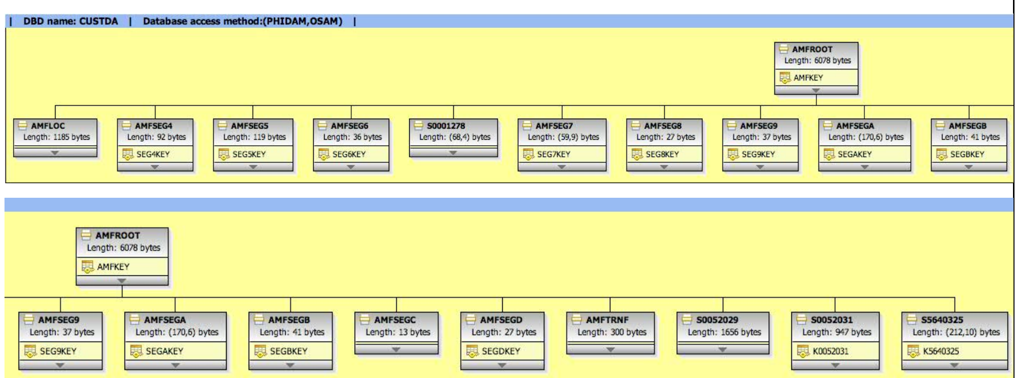
The full function with HALDB workload contains a mix of databases that are OSAM and VSAM with Hierarchical Direct Access Method (HDAM), Hierarchical Indexed Direct Access Method (HIDAM), Partitioned HDAM (PHDAM), and Partitioned Indexed HDAM (PHIDAM) databases. The workload contains 32 replicates. Each replicate consists of eight full-function databases. Half of the replicates are OSAM replicates while the other half are VSAM replicates as described below.

Table 2: Full function with HALDB Workload Database Description

<table>
<tr><th>Database Name</th><th>Database Overview</th></tr>
<tr><td>CUSTDA</td><td>CUSTDA is a HALDB customer database containing information for customers. A CUSTDA record is composed of the following segments:

- AMFROOT (root segment) – Contains information for customer
- Sixteen direct dependent segments containing dummy data

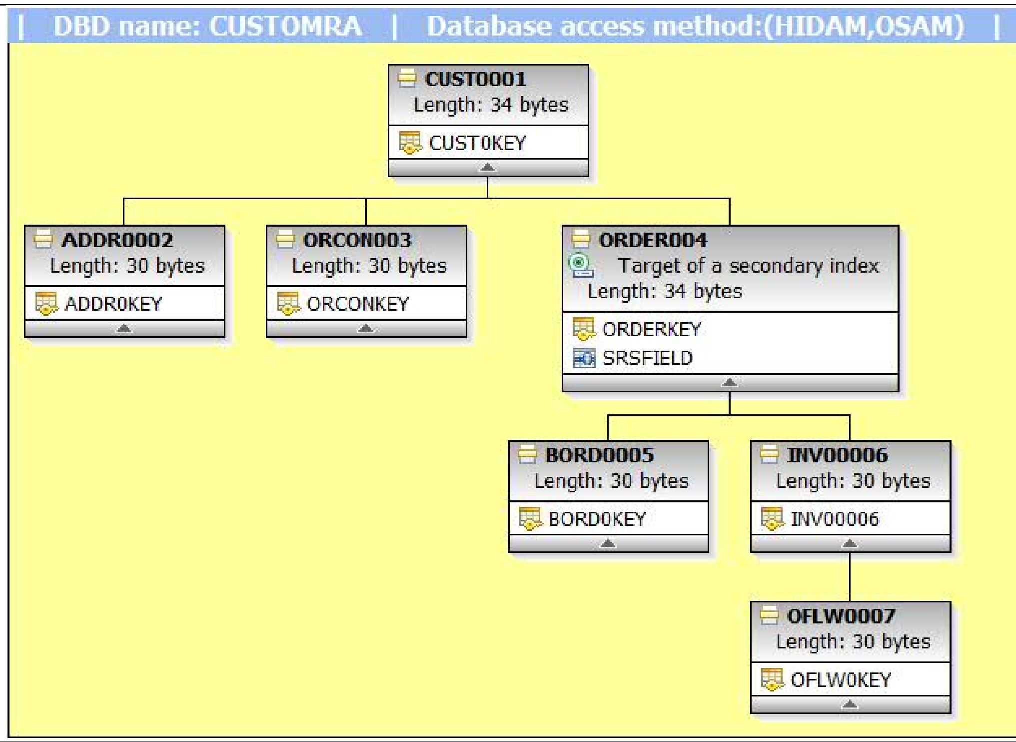
The following is a graphical view of the database layout:

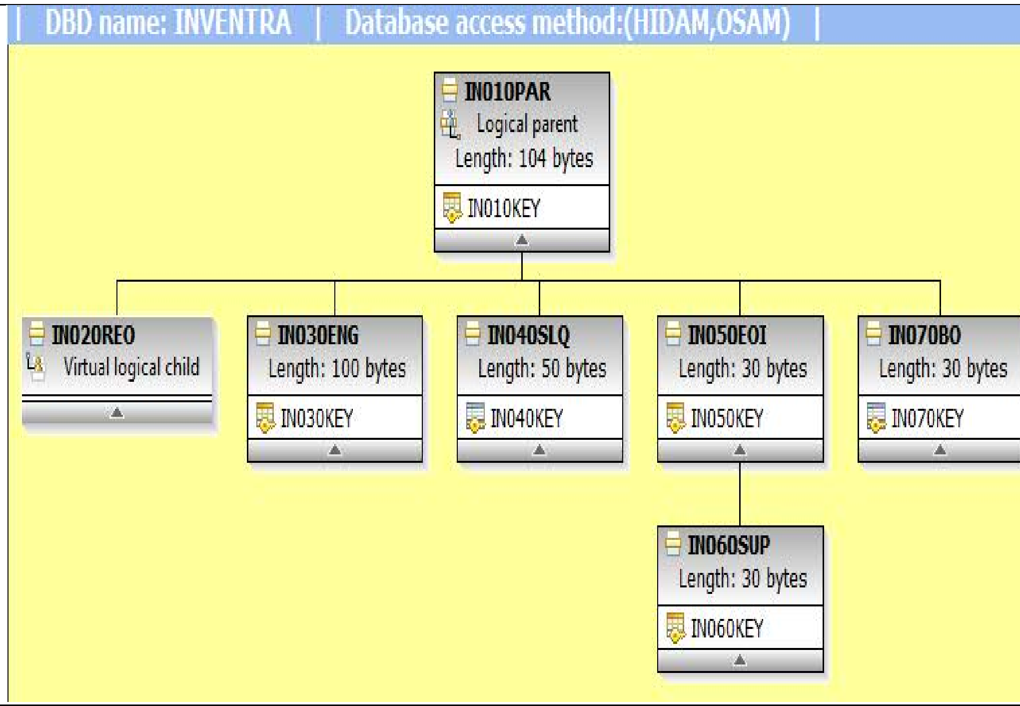


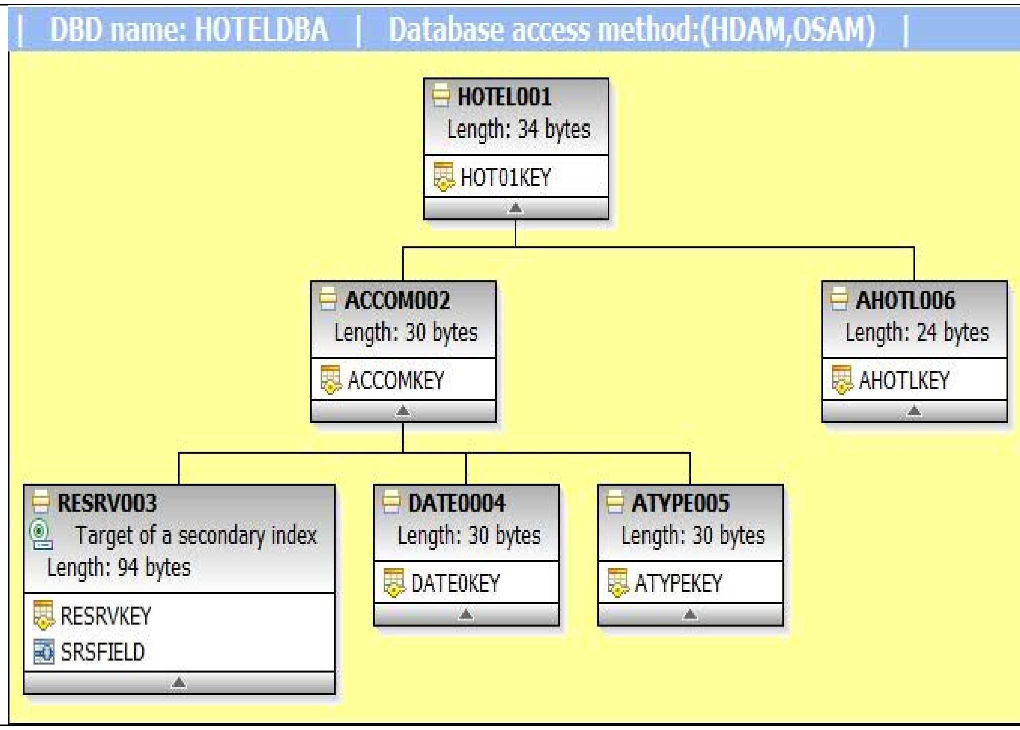
</td></tr>
<tr><td>CUSTOMRA</td><td>CUSTOMRA is a customer database containing information for the customer directory. A CUSTOMRA record is composed of the following segments:

- CUST001 (root segment key) - Contains information for customer
- ADDR002 (direct dependent segment key) - Contains information for address
- ORCOND003 (direct dependent segment key) - Contains information for order conditions
- ORDER004 (direct dependent segment key) - Contains information for order number
- BORD0005 (dependent segment key) - Contains information for back orders
- INV00006 (dependent segment key) - Contains information for invoice parts
- OFLW0007 (dependent segment key) - Contains information for overflow parts

The following is a graphical view of the database layout:</td></tr>
</table>

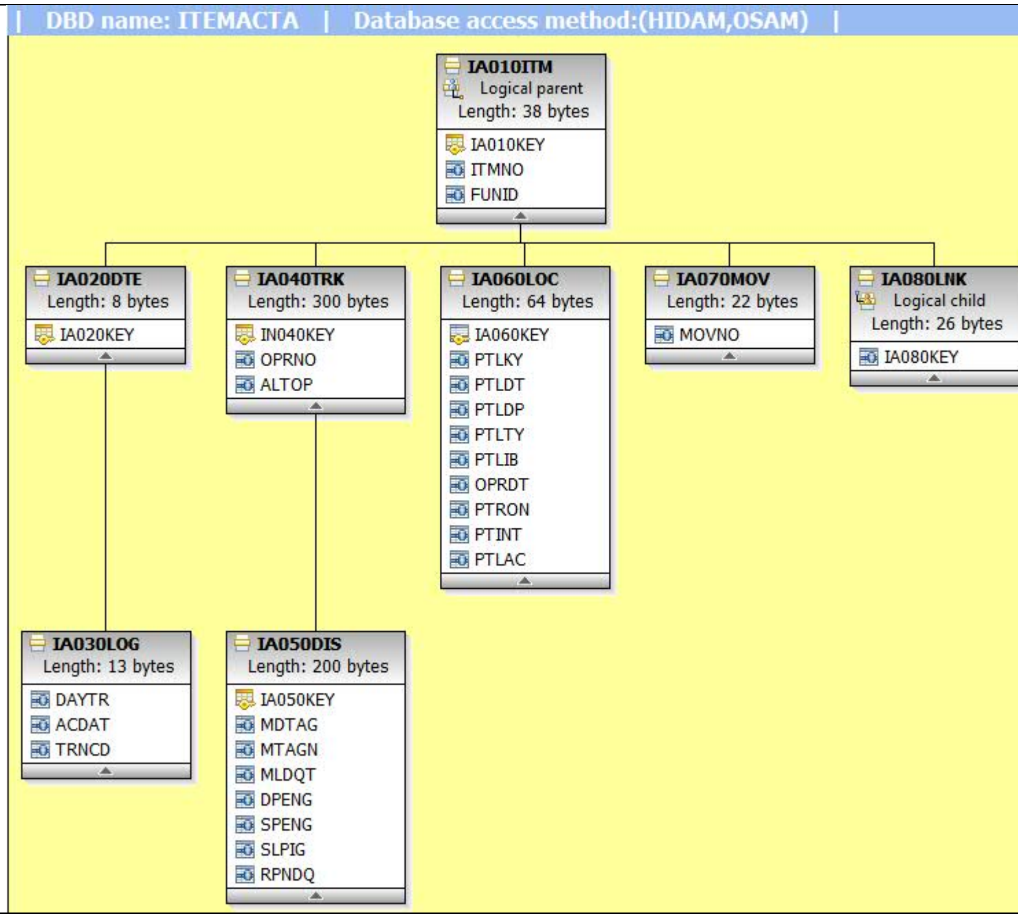
	
INVENTRA	INVENTRA is an inventory database containing information for all the orders and parts for the Stock Control application. A INVENTRA record is composed of the following segments: • IN010PAR (root segment key) - Contains information for parts • IN020REO (direct dependent segment key) - Contains information for reorders • IN030ENG (direct dependent segment key) - Contains information for engineering • IN040SLQ (direct dependent segment key) - Contains information for secondary location quantity • IN050EOI (direct dependent segment key) - Contains information for EOI • IN070BO (direct dependent segment key) - Contains information for back orders • IN060SUP (dependent segment key) - Contains information for suppliers The following is a graphical view of the database layout:

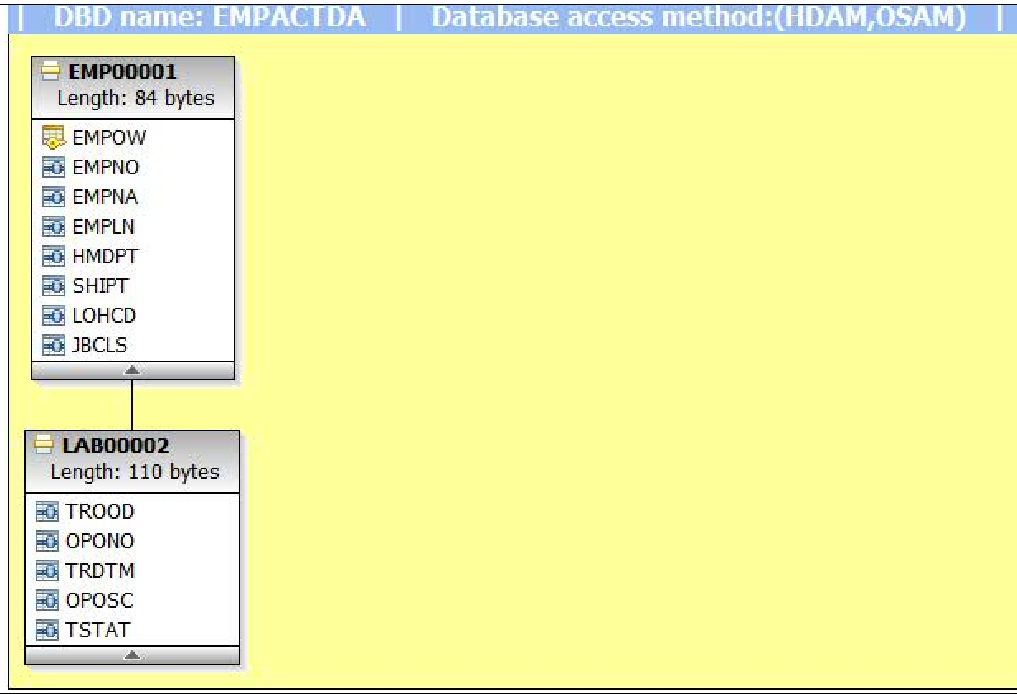
<table>
<tr><td></td><td>
</td></tr>
<tr><td>HOTELDBA</td><td>HOTELDBA is hotel reservation database pertaining information for the hotel reservation for specific requests like accommodation type, dates, and location. A HOTELDBA record is composed of the following segments:
• HOTEL001 (root segment key) - Contains information for hotel
• ACCOM002 (direct dependent segment key) - Contains information for accommodation type
• AHOTL006 (direct dependent segment key) - Contains information for alternate hotel
• RESRV003 (dependent segment key) - Contains information for reservations
• DATE0004 (dependent segment key) - Contains information for date
• ATYPE005 (dependent segment key) - Contains information for alternate type
The following is a graphical view of the database layout:</td></tr>
</table>

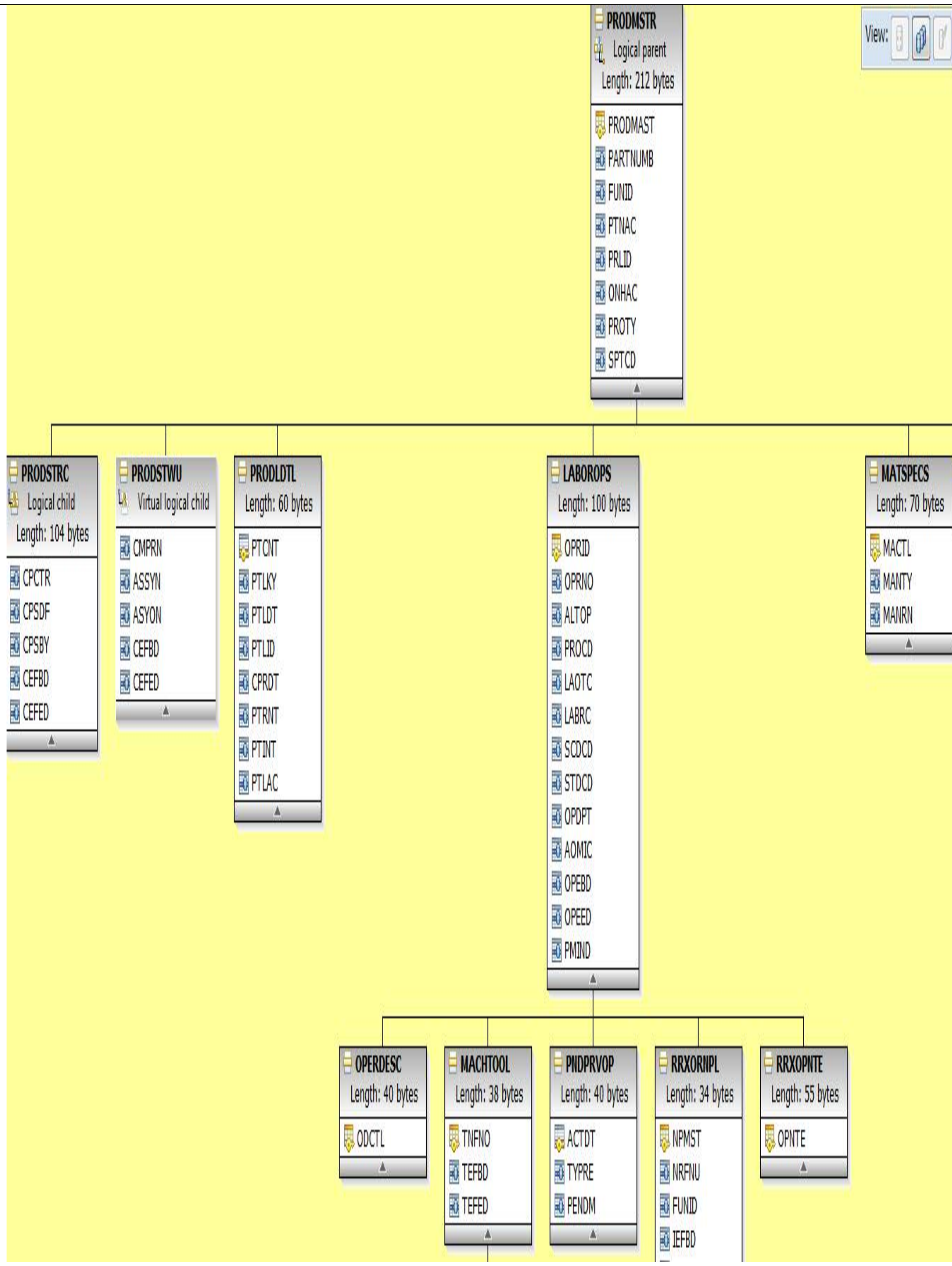
<table>
<tr><td></td><td>
</td></tr>
<tr><td>ITEMACTA</td><td>ITEMACTA is an item activity database containing information on receiving purchase order items, storage control and tracking material, as it moves through production operation units. A ITEMACTA record is composed of the following segments:

IA010ITM (root segment key) - Contains information for activity item number
IA020DTE (direct dependent segment key) - Contains information for activity date
IA060LOC (direct dependent segment key) - Contains information for location detail
IA080LNK (direct dependent segment key) - Contains information for purchase orders
IA070MOV (direct dependent segment key) - Contains information for move orders
IA040TRK (direct dependent segment key) - Contains information for track operations
IA030LOG (dependent segment key) - Contains information for activity log
IA050DIS (dependent segment key) - Contains information for disc material

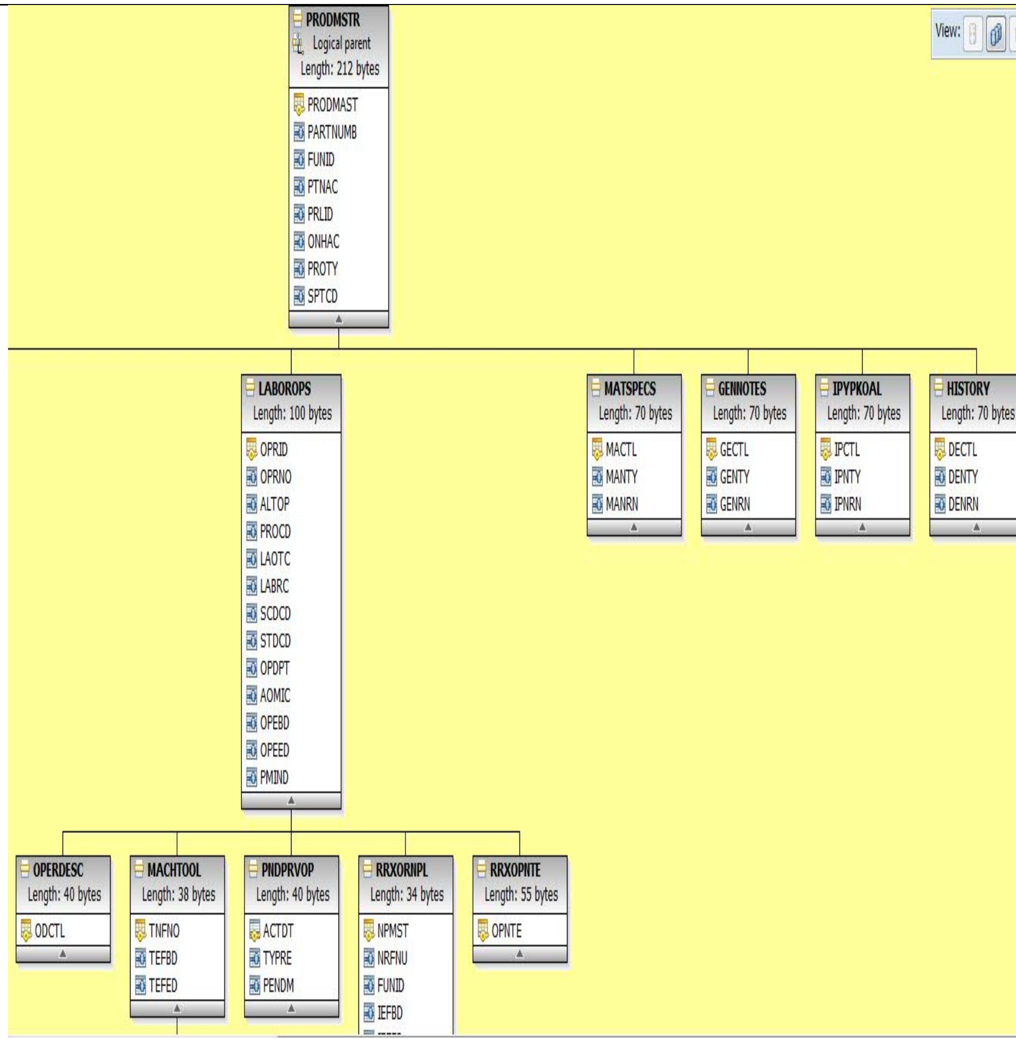
The following is a graphical view of the database layout:</td></tr>
</table>

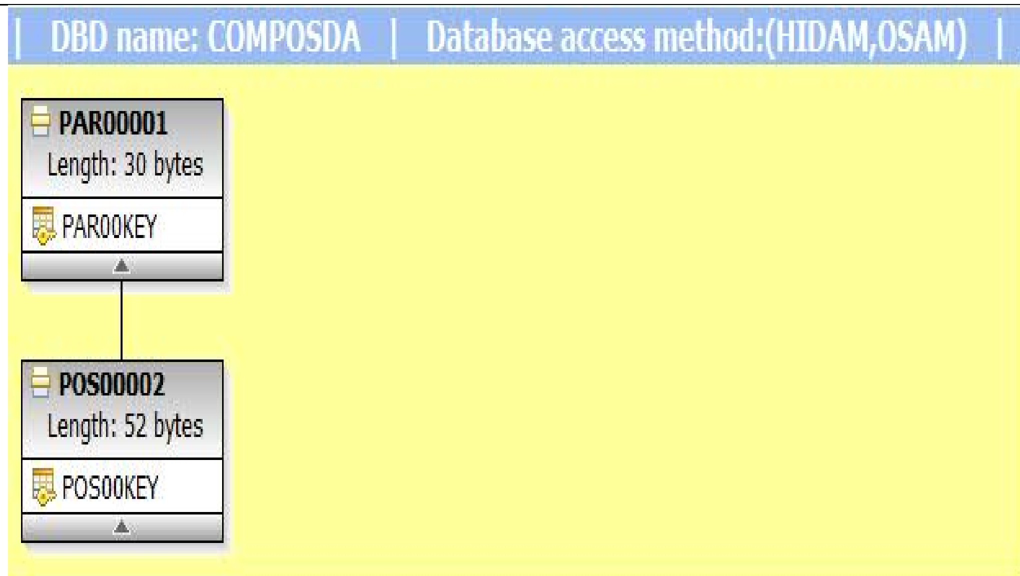
	
EMPACTDA	EMPACTDA is an employee activity database containing labor operations information for the employees, as the item is moved from one workstation to another. A EMPACTDA record is composed of the following segments: • EMP00001 (root segment key) - Contains information for employee number • LAB00002 (direct dependent segment key) - Contains information for labor status The following is a graphical view of the database layout:

	
ITEMMASA	PRODMSTR is a product item master database containing information for the product specifics. A PRODMSTR record is composed of the following segments: • PRODMSTR (root segment key) - Contains information for product master • PRODSTRC (direct dependent segment key) - Contains information for production structure • PRODSTWC (direct dependent segment key) - Contains information for production schedule • PRODLDTL (direct dependent segment key) - Contains information for product unit detail • LABOROPS (direct dependent segment key) - Contains information for labor operations • MATSPECS (direct dependent segment key) - Contains information for material specifications • GENNOTES (direct dependent segment key) - Contains information for general notes • IPYKOAL (direct dependent segment key) - Contains information for control • HISTORY (direct dependent segment key) - Contains information for history The following is a graphical view of the database layout divided into two graphics shown below:



View:
PRODMSTR
Logical parent
Length: 212 bytes
PRODMAST
PARTNUMB
FUNID
PTNAC
PRLID
ONHAC
PROTY
SPTCD
PRODSTRC
Logical child
Length: 104 bytes
CPCTR
CPSDF
CPSBY
CEFBD
CEFED
PRODSTWU
Virtual logical child
CMPRN
ASSYN
ASYON
CEFBD
CEFED
PRODLDTL
Length: 60 bytes
PTCNT
PTLKY
PTLDT
PTLID
CPRDT
PTRNT
PTINT
PTLAC
LABOROPS
Length: 100 bytes
OPRID
OPRNO
ALTOP
PROCD
LAOTC
LABRC
SCDCD
STDCD
OPDPT
AOMIC
OPEBD
OPEED
PMIND
MATSPECS
Length: 70 bytes
MACTL
MANTY
MANRN
OPERDESC
Length: 40 bytes
ODCTL
MACHTOOL
Length: 38 bytes
TNFNO
TEFBD
TEFED
PNDPRVOP
Length: 40 bytes
ACTDT
TYPRE
PENDM
RRXORNPL
Length: 34 bytes
NPMST
NRFNU
FUNID
IEFBD
RRXOPNTE
Length: 55 bytes
OPNTE

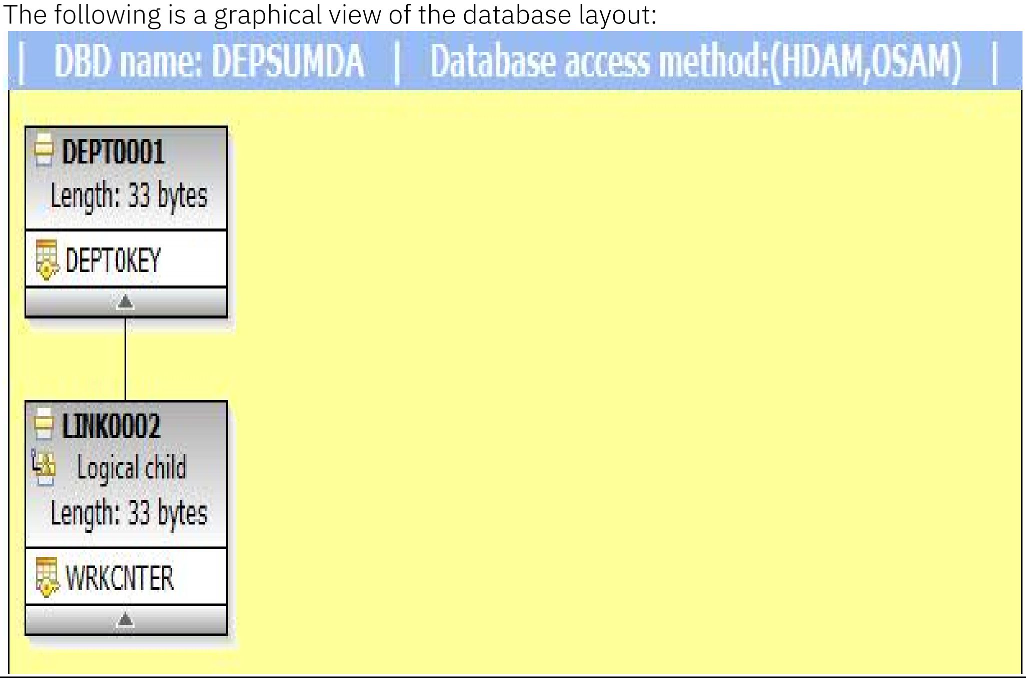
	
COMPOSDA	COMPOSDA is component position database containing information for the part numbers and their position data. A COMPOSDA record is composed of the following segments: • PAR00001 (root segment key) - Contains information for part number • POS00002 (direct dependent segment key) - Contains information for position data The following is a graphical view of the database layout:

<table>
<tr><td></td><td>


</td></tr>
<tr><td>DEPSUMDA</td><td>
DEPSUMDA is a department summary database containing information for the home department and work center. A DEPSUMDA record is composed of the following segments:

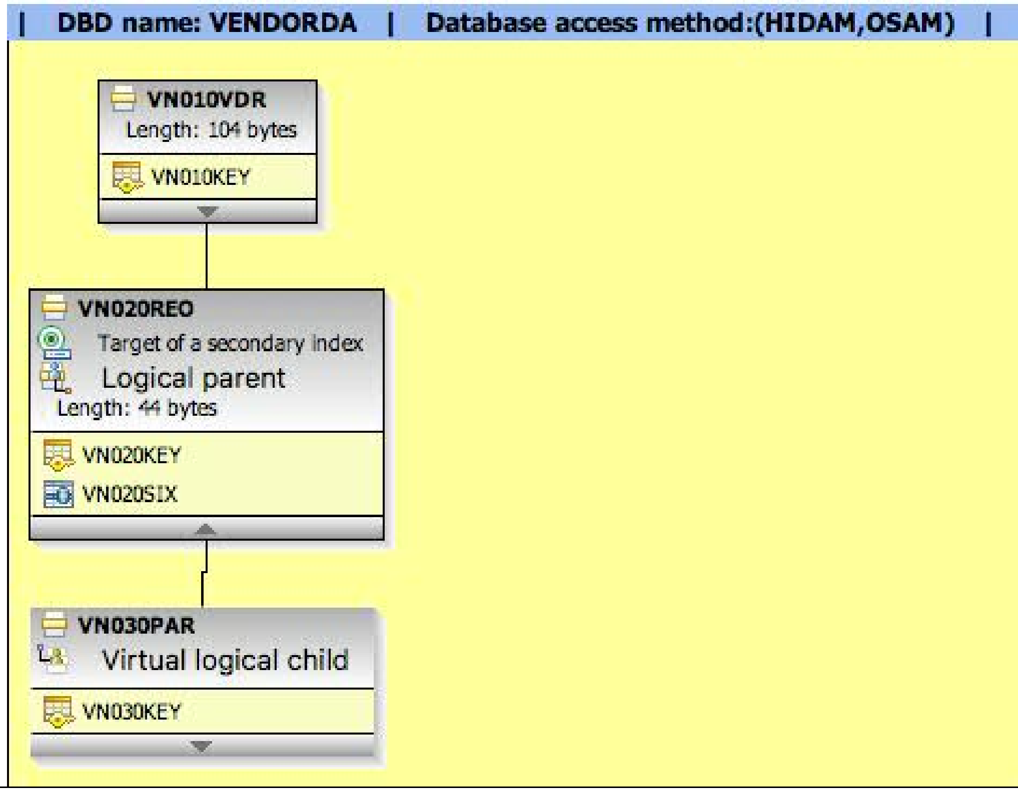
- DEPT0001 (root segment key) - Contains information for home department
- LINK0002 (direct dependent segment key) - Contains information for work center

The following is a graphical view of the database layout:



</td></tr>
<tr><td>VENDORDA</td><td>
VENDORDA is a vendor database containing information for all the vendors. VENDORDA record is composed of the following segments:

- VN010VDR (root segment key) - Contains information for vendor key
</td></tr>
</table>

	• VN020REO (logical parent segment key) - Contains information for reorder segment of vendor database via secondary index • VN030PAR (virtual logical child segment key) – Contains information for replacing a reorder segment The following is a graphical view of the database layout: 

4.1.2 Application - Transaction Description

The full function with HALDB workload runs a mix of transactions using teller system, inventory, hotel, and warehouse type transactions issuing read, replace, delete, and insert calls as described in Table 3 below. The application is written in COBOL and compiled with COBOL V6R3.

This is the complete list of transactions run by full-function with HALDB and CICS-full-function workload as described in Section 4.4.

Table 3: Full-function with HALDB Workload Transaction Description

Transaction Name	Transaction Overview
HR1	Hotel Reservation Application - Hotel Reservation Transaction: • Processes reservation requests for specific accommodations at specific locations for specific dates • If accommodation type is unavailable, alternate arrangements are recommended • Hotel Reservation Database accessed
HR2	Hotel Reservation Application – Reservation Segment Create Transaction: • Creates a third level reservation segment for the already held reservation • Includes new information in the dependent segment including arrival date, departure date, room type, bed type, number of persons, rate, and personal information • Hotel Reservation Database accessed
IT2	Inventory Tracking Application - Part Location and Inventory Inquiry Transaction: • Reports part location and available inventory information • Item Activity Database accessed
IT8	Inventory Tracking Application - Update in process Inventory Transaction: • Records the completion of labor operations on manufactured items and updates employee activity records • Item is tracked as it moves from workstation to workstation on manufacturing shop floor • Records two labor operations by a single employee on a single in-process item • Employee Activity Database, Item Activity, Product Item, Master and Department Summary Database accessed
OE1	Order Entry Application - Customer Inquiry Transaction: • Inquires the customer database indexed by customer number • The customer number, if not directly known, is looked up in a customer directory • The output consists of customer name, ship to and bill to address, possible discounts and current credit • Conversational transaction. Data is stored in the SPA for further inquiry • Customer Database (CUSTOMRA) accessed
OE2	Order Entry Transaction - Customer Change: • Makes changes to the customer database • Mostly affects the root and second level address segment • Input consists of altered customer information from the customer inquiry step • Conversational transaction • Customer Database (CUSTOMRA) accessed
OE4	Order Entry Transaction - Parts Inquiry: • Reads a list of part numbers and quantities requested from inputting terminal • Makes inquiry into the status of each of these

	• Conversational transaction • Inventory database accessed
OE5	Order Entry Transaction - Parts Processing: • Makes actual updates to the inventory database when filling an order and producing an invoice • Updates the quantity on hand for each of the parts and creates a back-order entry if required • Also affects customer database, as order must be produced under appropriate customer • Final step for Order Entry conversational transactions • Output consists of terminal response and an invoice • Inventory and customer database (CUSTOMRA) accessed
PS2	Production Specification Application - Bill of Materials Transaction: • Bills of Materials for design level components • Product Item Master and Component Position Database accessed
PS3	Production Specification Application - Labor Operation/Incentive Standards for Parts Transaction: • Inquires the labor operations and incentive standards for a given part • Product Item Master Database accessed
SC6	Stock Control Application - Supplier Delete Transaction: • Delete obsolete suppliers from the Inventory database • Input is supplier's number and part number. Record may be deleted if there are no longer any orders pending • Database maintenance function • Inventory Database accessed
TS1	Teller System Application - Data Entry Transaction: • This is a HALDB transaction that does inserts and deletes • Customer Database (CUSTDA) accessed

4.1.3 Application - Workload Distribution

The full-function with HALDB workload uses transactions with the following execution distributions from the asynchronous driver as shown in Table 4 below.

Table 4: Full-function with HALDB Workload Transaction Distribution

Transaction	Distribution
HR1	6,56%
HR2	6.56%
IT2	6.56%
IT8	6.56%
OE1	6.56%
OE2	6.56%

OE4	6.56%
OE5	6.56%
PS2	18.03%
PS3	16.39%
SC6	6.56%
TS1	6.56%

Order Entry (OE) transactions are conversational, Program-to-Program switch transactions. OE1 calls OE2 which initiates OE4 which finally calls OE5.

4.2 Data Sharing Full Function with High Availability Large Database and Shared Message Queues Workload

IMS provides the ability for multiple IMS systems in a Parallel Sysplex® environment to share a single set of message queues held in structures within a Coupling Facility (CF). The IMS Shared Message Queue (SMQ) uses the Common Queue Server (CQS), which is a generalized server that manages objects on CF structures, such as list or message queue structures. CQS receives, maintains, and distributes data objects from shared queues on behalf of its clients.

The data sharing full function (DSFF) with HALDB and SMQ contain the same full-function databases and transactions described in Section 4.2; however, it uses IMS shared queues (versus local queues) for its message processing.

The workload performs updates to a variety of OSAM and VSAM FF databases and uses Internal Resource Lock Manager (IRLM) as the data sharing lock manager.

The databases were defined to Database Recovery Control (DBRC) using block-level data sharing (SHARELVL (3)) to support data sharing for the SMQ workload.

4.2.1 Database Description

The DSFF with HALDB and SMQ workload consists of the same databases as described in Section 4.1.1.

4.2.2 Application - Transaction Description

The DSFF with HALDB and SMQ workload consists of the same transactions as described in Section 4.1.2.

4.2.3 Application - Workload Distribution

The DSFF with HALDB and SMQ workload runs the same transaction distributions as described in Table 4 in Section 4.1.3.

4.3 Fast Path (FP) Banking Workload

Fast Path databases and message processing provide high performance processing in a high-available data environment for IMS applications. The Fast Path banking workload represents application functions that might comprise the high-volume online workload in a credit card environment. Its capabilities include credit card validation, credit limit check, debit/credit account, and lost/stolen card reporting.

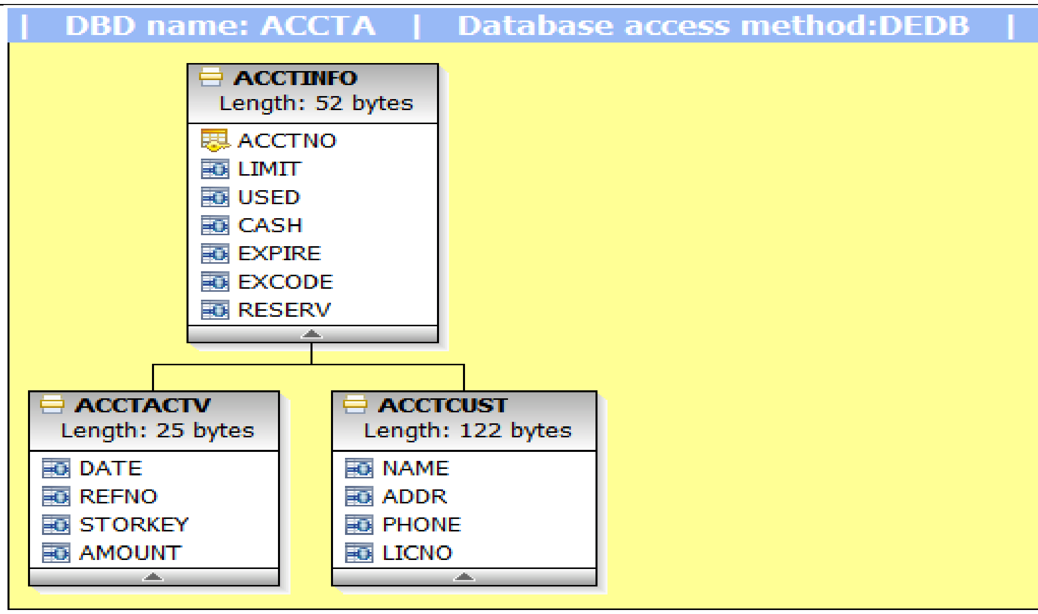
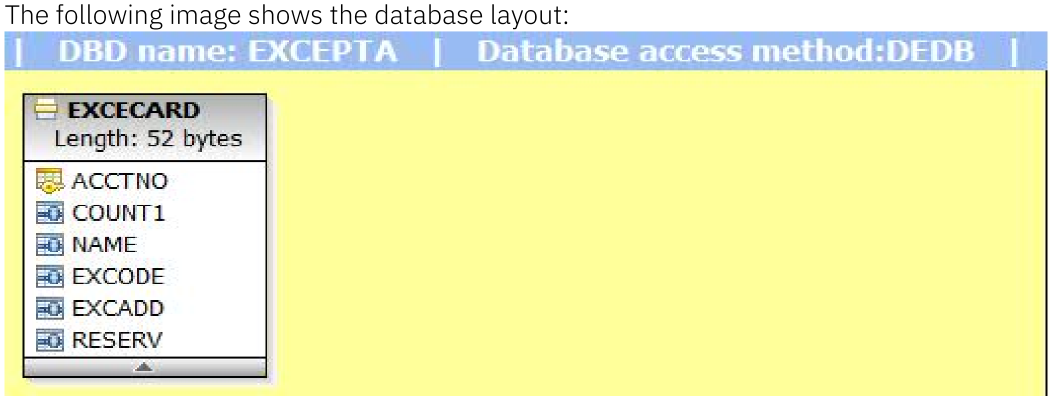
This section provides details about the database description, application transactions, and workload transaction distribution.

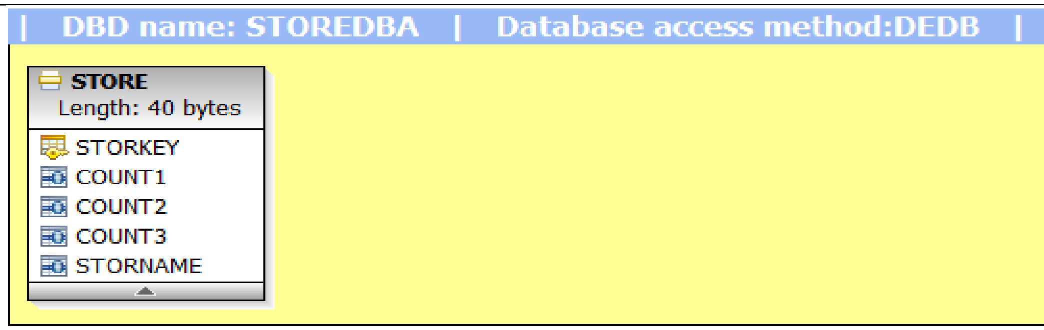
4.3.1 Database Description

The Fast Path banking workload consists of three Fast Path Data Entry Databases (DEDB) as described in Table 5.

Table 5: Fast Path Workload Database Description

Database Name	Database Overview
ACCTA	The Account Database (DEDB) is a customer account database containing general information. An ACCTA record is composed of the following segments: • ACCTINFO (root segment) - Contains information for account information. Major fields include account number, credit limit, and used credit • ACCTCUST (direct dependent segment) - Contains information for account customer. Major fields include customer name and address • ACCTACTV (direct dependent segment) - Contains information for account activity. One ACCTACTV segment instance exists for every debit/credit against an account. Major fields include the amount of the debit/credit, store code number, and date of transaction This database consists of 480 DEDB Area Data sets. The following image shows the database layout:

<table>
<tr>
<td></td>
<td>
</td>
</tr>
<tr>
<td>EXCEPTA</td>
<td>The Exception Card Database (DEDB) is an exception card database containing records for lost, stolen, or otherwise invalid cards. This database is periodically maintained (offline) with EXCADD cards plus other updates. An EXCEPTA record consists of the following segment:
• EXCECARD (root segment) - Contains information for exception card. Major fields include account number, customer name, and number of attempted uses
This database consists of one DEDB area data set.
The following image shows the database layout:

</td>
</tr>
<tr>
<td>STOREDBA</td>
<td>Store Database (DEDB) is a database containing a record for each “establishment/retailer” subscribing to the credit card service. A STOREDB record consists of the following segment:
• STORE (root segment) - Contains information for store and about the establishment. Major fields include store code number, store name, and number (by type) of each transaction invoked
This database consists of 16 DEDB area data sets.
The following image shows the database layout:</td>
</tr>
</table>



4.3.2 Application - Transaction Description

The Fast Path banking workload runs a mix of transactions in an online credit card environment executing read, replace, and insert calls.

The Fast Path banking workload includes four unique transactions with varying levels of processing. All transactions run in IFP regions and are Fast Path only; that is, they are Fast Path Expedited Message Handler (EMH) messages and only access FP DEDBs. The transactions do not access any full-function DL/I databases or issue calls to the IMS message queue. The transactions provide an online update capability with full integrity and recovery facilities for both the Fast Path databases and the Fast Path output response messages. Each transaction starts with an 80-byte input message and replies with an 83-byte output message. The application is written in COBOL and compiled with COBOL V6R3.

The IMS transactions CCCK, CLCK, DEBIT, and CREDIT would normally be issued by establishments subscribing to the credit card service. Counters in the STOREDBA database's store record tally of the number of DEBIT and CREDIT transactions issued. Table 6 provides an overview of the processing done by each of these transactions.

Table 6: IMS Fast Path Workload Transaction Overview

Transaction Name	Transaction Overview
CCCK	Credit Card Authorization Check: • Search the Exception Card database to see if this card has been reported as lost or stolen • If necessary, also search the Added Exception Card database • If the card was found, update the "attempted uses" field in the corresponding record • Increment the transaction count in the Store Database

	• Return a message indicating the outcome of the search ("authorization ok" or "authorization denied") to the requesting client Note: Usually there is no I/O since most cards are in good standing, and the Added Exception Card database will not need to be searched. The vast majority of transactions processed will stop at first bullet point with "authorization ok" returned in last bullet point.
CLCK	Credit Limit Check: • Fetch the Account Database root and check the transaction amount against the available credit • Increment the transaction count in the Store Database • Return a message authorizing or denying the purchase to the requesting client
DEBIT/CREDIT	Debit/Credit: - Fetch the Account Database root and update the balance - Insert a direct dependent under this root to journal the account activity - Increment the transaction count in the Store Database - Return "transaction complete" message to the requesting client Note: Each transaction requires DEDB read and an DEDB write.

4.3.3 Application - Workload Distribution

The transactions in the IMS Fast Path banking workload have the following execution distributions as shown in Table 7.

Table 7: IMS Fast Path Workload Transaction Distribution

Transaction	Distribution
CCCK	33.34%
CLCK	33.34%
DEBIT/CREDIT	33.34%

4.4 Customer Information Control System (CICS) - IMS DBCTL Workload

The Customer Information Control System (CICS) transaction server workload was tested using the following IMS databases:

- CICS FF (retail/warehouse) workload as described in Section 4.1
- CICS FP (banking) workload as described in Section 4.3

The CICS FF workload uses transactions with the following execution distributions as shown in Table 8.

Table 8: CICS Full-Function Workload Transaction Distribution

Transaction	Distribution
HR1	10%
HR2	10%
IT2	10%
IT8	10%
PS2	25%
PS3	25%
SC2	5%
SC6	5%

The CICS FP workload uses transactions with the following execution distributions as shown in Table 9.

Table 9: CICS Fast Path Workload Transaction Distribution

Transaction	Distribution
CCCK	9.17%
CLCK	30.28%
DEBIT	30.28%
CREDIT	30.28%

4.5 IMS TM-Db2 with IBM Relational Warehouse Workload (IRWW) - Full PL/I

The IMS-Db2 IRWW relational database is based on a retail type of environment. The IRWW database resides on a Db2 for z/OS database. The IMS transaction executes in 240 MPP regions accessing the Db2 warehouse database through the External Subsystem Attach Facility (ESAF).

4.5.1 Database Description

The workload consists of seven transactions with each transaction consisting of one-to-many SQL statements and containing inventory stock warehouses and sales districts. This workload has nine tables with five of the nine tables being in the partitioned table space as described below:

Table 10: IMS TM-Db2 Database Description

Table Name	Table Overview
CUSTOMER	CUSTOMER is 10 parts with index/part + one additional index and has 21 columns per row
DISTRICT	DISTRICT is partition-by-growth with one unique index and has 11 columns per row
HISTORY	HISTORY is partition-by-growth without index and has eight columns per row
ITEM	ITEM is partition-by-growth with one unique index and has four columns per row
NEWORDERS	NEWORDERS is 10 parts with index/part and has three columns per row
ORDERLINE	ORDERLINE is 10 parts with index/part and has 10 columns per row
ORDERS	ORDERS is 10 parts with index/part and has eight columns per row
STOCK	STOCK is 10 parts with index/part and has 17 columns per row
WAREHOUSE	WAREHOUSE is partition-by-growth with one unique index and has nine columns per row

4.5.2 Application - Transaction Description

The IRWW workload runs a mix of transactions using retail warehouse type transactions executing select, fetch, update, insert, and delete calls.

Table 11: TM-Db2 Application Transaction Description

Transaction Name	Transaction Description
Delivery	Performs various SELECTs, UPDATEs, and DELETEs in support of the delivery of a group of orders
New Order	Performs various SELECTs, FETCHs, UPDATEs, and INSERTs in support of the new customer
Order Status	Performs various SELECTs, FETCHs in support of providing the status of an order
Payment	Performs various SELECTs, FETCHs, UPDATEs, and INSERTs in support of the received customer payment
Price Change	Performs an UPDATE in support of changing the price of an item
Price Quote	Performs a SELECT in support of providing the price of a set of items
Stock Inquiry	Performs a JOIN and various SELECTs in support of providing the current stock level of an item

4.5.3 Application - Workload Distribution

The transactions in the IRWW have the following execution distribution as shown in Table 12.

Table 12: TM-Db2 Workload Distribution

Transaction	Distribution
Delivery	2%
New Order	22%
Order Status	24.5%
Payment	22%
Price Change	0.5%
Price Quote	25%
Stock Inquiry	4%

4.6 z/OS Connect with IMS Service Provider (SP) Workload

The IBM z/OS Connect is an integrated solution that enables developers to merge business applications into today's growing mobile, cloud, and hybrid cloud application ecosystems. The z/OS Connect combines IBM and industry state of the art technologies to deliver an effective, intuitive solution for defining services and APIs to access your IMS assets using industry standard REST technology.

The following three components in z/OS Connect provide API solutions for IMS:

- The z/OS Connect
- The IMS Service Provider
- z/OS Connect API toolkit

Each of these components integrates seamlessly to provide a fast and reliable experience for developers as they build applications for mobile and cloud use cases where speed to market is critical.

This workload focuses on stressing the IMS SP code and consists of message-only transactions without any database activity.

4.6.1 Application - Transaction Description

The workload consists of 25 message-only transactions with transaction codes VARTX001 to VARTX025. All transactions run in MPP regions. Figure 3 shows the transaction attributes from issuing IMS command /DIS TRAN ALL below.

TRAN	CLS	ENQCT	QCT	LCT	PLCT	CP	NP	LP	SEGSZ	SEGNO	PARLM	RC
VARTX001	1	0	0	65535	65535	1	1	1	0	0	0	0
VARTX002	2	0	0	65535	65535	1	1	1	0	0	0	0
VARTX003	3	0	0	65535	65535	1	1	1	0	0	0	0
VARTX004	4	0	0	65535	65535	1	1	1	0	0	0	0
VARTX005	5	0	0	65535	65535	1	1	1	0	0	0	0
VARTX006	6	0	0	65535	65535	1	1	1	0	0	0	0
VARTX007	7	0	0	65535	65535	1	1	1	0	0	0	0
VARTX008	8	0	0	65535	65535	1	1	1	0	0	0	0
VARTX009	9	0	0	65535	65535	1	1	1	0	0	0	0
VARTX010	10	0	0	65535	65535	1	1	1	0	0	0	0
VARTX011	11	0	0	65535	65535	1	1	1	0	0	0	0
VARTX012	12	0	0	65535	65535	1	1	1	0	0	0	0
VARTX013	13	0	0	65535	65535	1	1	1	0	0	0	0
VARTX014	14	0	0	65535	65535	1	1	1	0	0	0	0
VARTX015	15	0	0	65535	65535	1	1	1	0	0	0	0
VARTX016	16	0	0	65535	65535	1	1	1	0	0	0	0
VARTX017	17	0	0	65535	65535	1	1	1	0	0	0	0
VARTX018	18	0	0	65535	65535	1	1	1	0	0	0	0
VARTX019	19	0	0	65535	65535	1	1	1	0	0	0	0
VARTX020	20	0	0	65535	65535	1	1	1	0	0	0	0
VARTX021	21	0	0	65535	65535	1	1	1	0	0	0	0
VARTX022	22	0	0	65535	65535	1	1	1	0	0	0	0
VARTX023	23	0	0	65535	65535	1	1	1	0	0	0	0
VARTX024	24	0	0	65535	65535	1	1	1	0	0	0	0
VARTX025	25	0	0	65535	65535	1	1	1	0	0	0	0

Figure 3: IMS Service Provider Workload Transaction Attributes

4.6.2 Application - Message Structures

The application was designed to receive and return messages with variable lengths based on the incoming message's field values. The evaluations in this paper focused on an I/O message size of 1 KB only. The I/O messages contain simple character fields.

The IMS SP provides data transformation between JavaScript Object Notation (JSON) and the binary format that the IMS transaction expects. The message structure was imported from the COBOL copybook structure using the z/OS Connect API toolkit. At run time, the data and message metadata structures are dynamically inspected and converted.

4.6.3 Application - Workload

A stand-alone Java application was used to drive the workload by sending REST API requests across several threads to simulate multiple clients. Each request calls a REST API using HTTPS GET and randomly invokes one of the 25 services deployed on the z/OS Connect server.

4.7 Java Message Processing (JMP) Workload

JMP regions are similar to MPP regions except they schedule Java programs rather than native applications such as COBOL or PL/I. In the PSB source associated with the Java program, the option LANG=JAVA must be specified. A JMP application is started when there is a message in the queue for the JMP application, and IMS schedules the message to be processed. JMP applications, like MPP applications, are executed through transaction codes submitted by users at terminals and from other applications. Each transaction code represents a transaction that the JMP application processes.

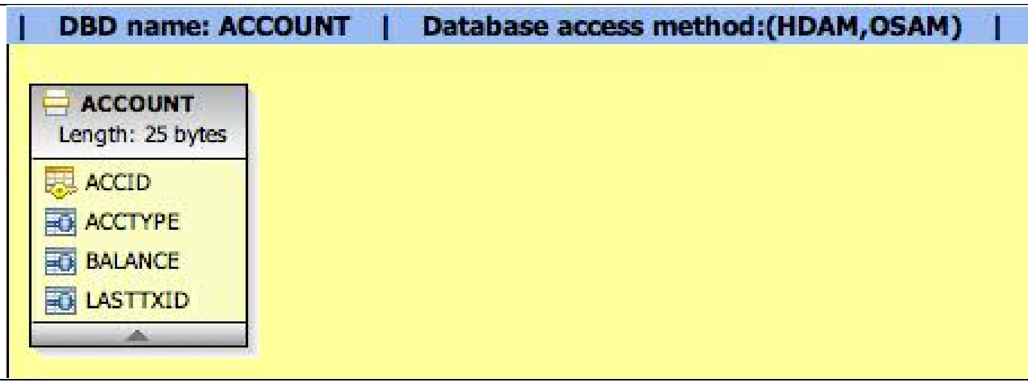
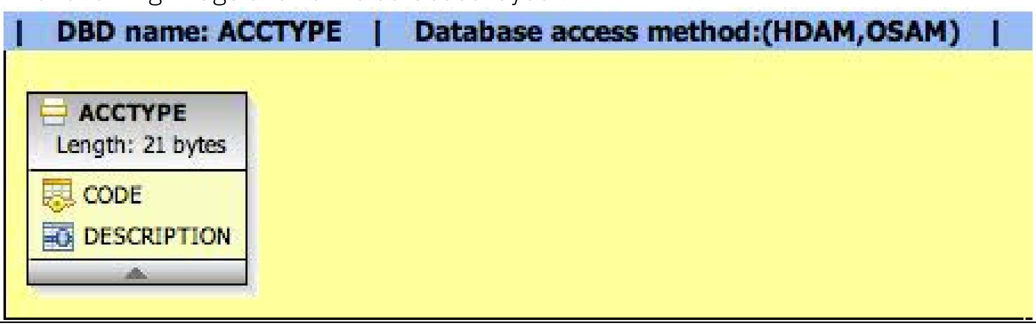
This section provides details about the database description, application transactions, and workload distributions.

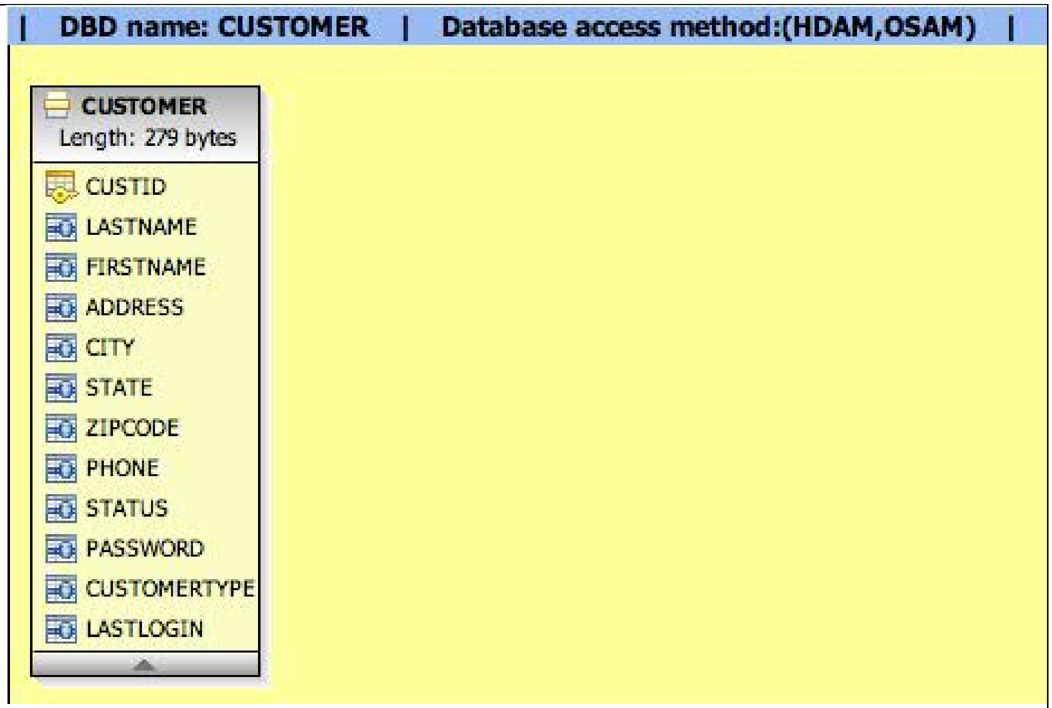
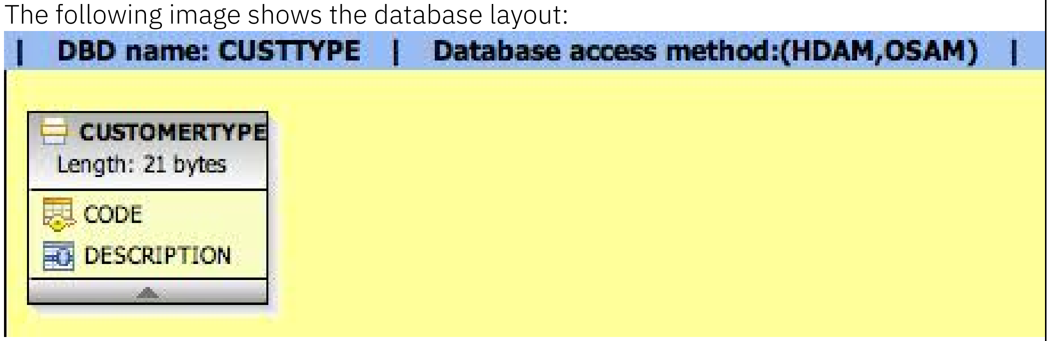
4.7.1 Database Description

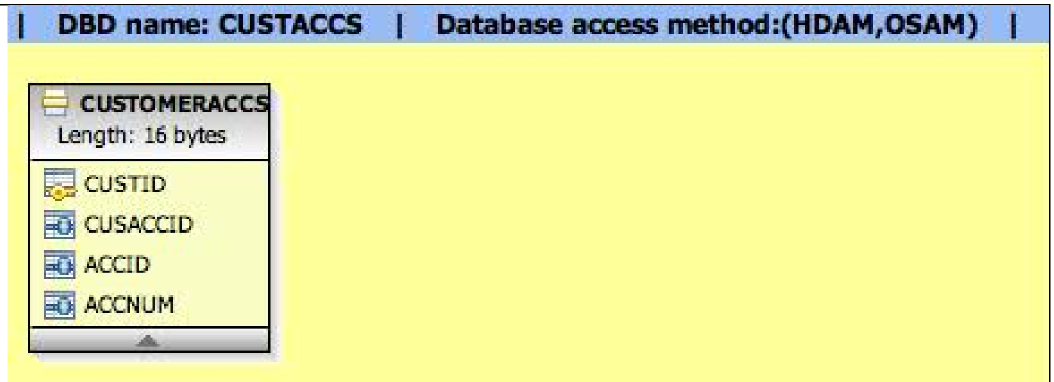
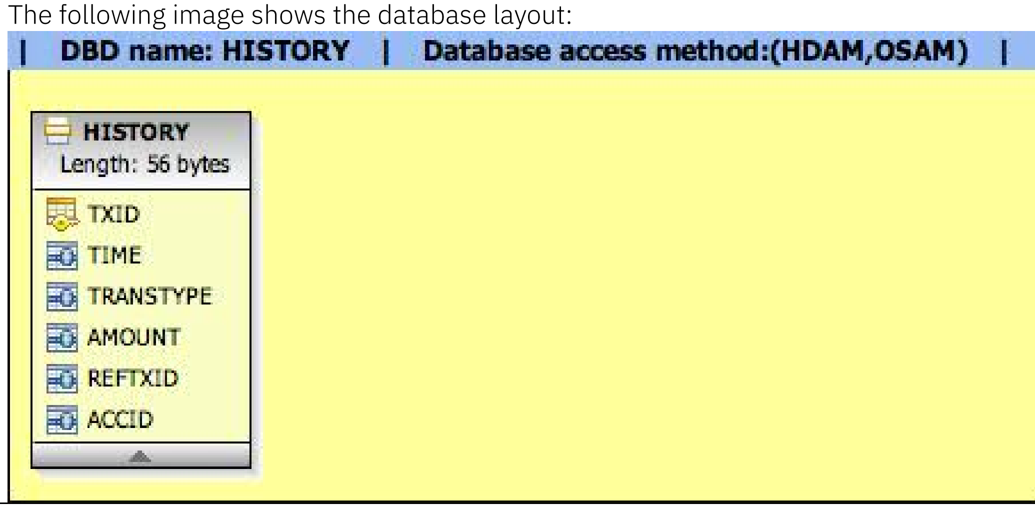
The JMP workload consists of six databases described in Table 13.

Table 13: JMP Workload Database Description

Database Name	Database Overview
ACCOUNT	The Account Database is a customer account database containing general information. It contains one root segment: • ACCOUNT (root segment) - Contains information for all customer accounts. Major fields include Account ID, Account type, Balance, and Last transaction ID The following shows the database layout:

<table>
<tr><td></td><td>
</td></tr>
<tr><td>ACCTYPE</td><td>The Account Type Database records all types of accounts in the account database. It contains one root segment:
ACCTYPE (root segment) - Contains information for the type of customer account. Major fields include Code and Description of the account
The following image shows the database layout:

</td></tr>
<tr><td>CUSTOMER</td><td>Customer Database is a database containing a record for all customers. It contains one root segment:
CUSTOMER (root segment) - Contains information for all customers. Major fields include customer ID, Last name, First name, Address, City, State, Zip code, Phone, Status, Password, Customer type, and Last login
The following image shows the database layout:</td></tr>
</table>

	
CUSTTYPE	The Customer Type Database records all types of customers in the customer database. It contains one root segment: • CUSTTYPE (root segment) - Contains information for all customer types. Major fields include Customer code and Description The following image shows the database layout: 
CUSTACCS	The Customer Account Segment Database records all types of customers in the customer account database. It contains one root segment: • CUSTACCS (root segment) - Contains information for all customer accounts. Major fields include Customer ID, Customer account ID, and Account number The following image shows the database layout:

	
HISTORY	The History Database records all types of customers transactions, time, Transaction types, Amount and Reference transaction IDs in History database. It contains one root segment: • HISTORY (root segment) - Contains information for all transactions. Major fields include Transaction ID, Time, Transaction type, Amount and Account ID The following image shows the database layout: 

4.7.2 Application - Transaction Description

The JMP workload runs a mix of transactions in an online credit card environment executing read, replace, and insert calls.

The IMS transactions FBTRAN, FBLOGIN, FBLOGOUT, FBACSUM, FBGCUDAT and FBSCUDAT are issued by establishments doing credit, account inquiries, logging in and logging out a customer. Table 14 provides an overview of the processing done by each of these transactions.

Table 14: JMP Workload Transaction Description

Transaction Name	Transaction Overview

FBTRAN	Friendly Bank Credit or Debit Check: • Fetch the Account Database root and check the transaction amount against the available credit • Insert a record and increment the transaction count in the History Database • Update Account Summary and Balance
FBLOGIN	Friendly Bank Account Login: • Perform login for customer ID into the account, verify password match and check if already logged on • Update last login information
FBLOGOUT	Friendly Bank Account Logout: • Perform logout for customer ID from the account • Update customer status
FBACSUM	Friendly Bank Account Summary: • Fetch the customer account information summary like balance, account type, and account ID
FBGCUDAT	Friendly Bank Get Customer Data: • Fetch the customer information like first name, last name, and address

4.7.3 Application - Workload Distribution

The transactions in the JMP workload have the following execution distributions as shown in Table 15.

Table 15: JMP Workload Database Distribution

Transaction	Distribution
FBTRAN	20%
FBLOGIN	20%
FBLOGOUT	20%
FBACSUM	20%
FBGCUDAT	20%

4.8 Open Database (ODBM) Workload

The IMS Open Database solution provides distributed access to IMS database resources driving industry standard and open technology into IMS. The distributed nature is twofold:

- At an IMSplex level, allows cross-LPAR access to any IMS database in the IMSplex
- At a pure distributed level, allows open platforms access to IMS resources directly by using industry standard interfaces

This is accomplished by three main components:

- Client libraries implementing the industry standard interfaces and protocols
- IMS Connect to process the distributed requests
- ODBM address space to process database access requests

4.8.1 Database Description

IMS Open Database consists of six full-function OSAM databases as described in Section 4.7.1, in Table 13: JMP Workload Database Description.

4.8.2 Application - Workload Distribution

The workload runs a mix of transactions in an online credit card environment executing read, replace, and insert calls as described in Section 4.7.3.

5 Measurement Methodology

The IMS performance evaluation cycle, as shown in Figure 4, is a repetitive process where a test environment is created and/or customized for a specific measurement, performance tests are run, and data is analyzed. Depending on the analyzed results, changes are made in the environment, if applicable, and the entire test process is repeated. All testing is done using an isolated and stable environment to produce consistent and repeatable performance measurement results.

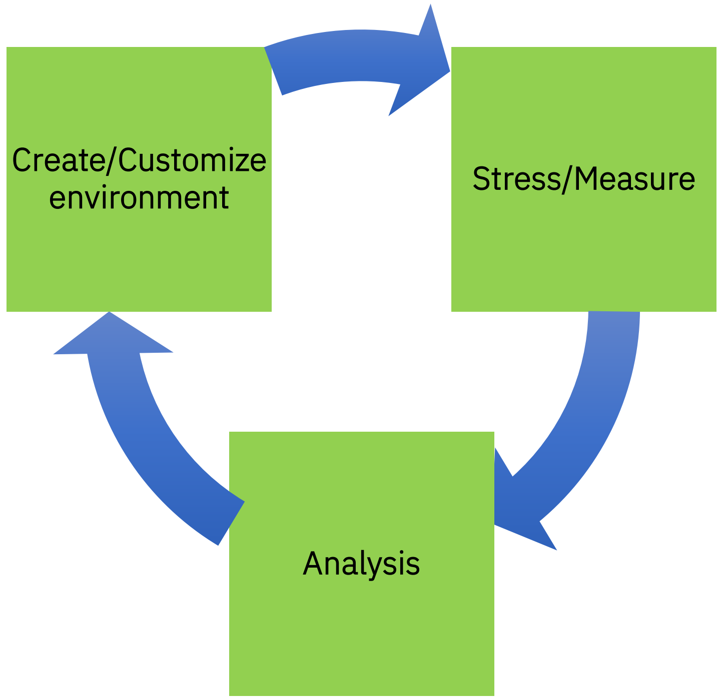


Figure 4: IMS Performance Evaluation Cycle

The performance test environment for each measurement evaluation described in this paper involves a specific system configuration of the number of LPARs and GPs that were active.

The system configuration is documented in the introduction of each measurement evaluation and kept constant when comparing z16 against z15.

A series of scaling tests were run to compare z16 against z15 at various CPU percent busy values ranging from 10% to 80%. For each workload, an approximate 80% (+/- 4%) CPU busy measurements were used to compare the z16 with z15.

5.1 Pre-measurement Procedure

The following general procedure was used to set up the measurement environment before the start of the measurement procedure comparing z16 against z15:

1. Restore the IMS database data sets.
2. Allocate IMS system data sets including Online Log Data sets (OLDS), Write Ahead Data sets (WADS), and Recovery Control data sets (RECONs).

3. Initialize IMS RECONs for database recoverability.
4. Start Structured Call Interface (SCI), Operations Manager (OM), and IRLM.
5. Initialize and cold start IMS.
6. Start the MPP/IFP/JMP regions depending on the measurement.
7. Start other workload-specific address spaces (for example, IMS Connect, CICS, Db2, and z/OS Connect) as required.

5.2 Measurement Procedure

After the databases are restored and environment is initialized, the following general steps were taken to measure the performance of a specific workload. The measurement procedure captures key performance data about the overall z/OS system as well as data that is specific to IMS. The following procedure was used to capture the measurement:

1. Initialize and start all Java or asynchronous drivers to begin driving transaction requests through IMS Connect over TCP/IP socket connections.
2. Ramp up the workload by adjusting the delay time (delay time specifies the time the asynchronous driver simulated “client” waits between each transaction invocation) for asynchronous drivers.
3. Start the Resource Management Facility (RMF™) Monitor I and III in a two-minute RMF data capture interval.
4. Issue the /SWITCH OLDS command to force an IMS OLDS switch.
5. Issue the /CHECKPOINT STATISTICS command to request that IMS performance records are created and written to the IMS log.
6. Wait for the two-minute measurement interval.
7. Issue the /CHECKPOINT STATISTICS command to request that IMS performance records are created and written to the IMS log.
8. Issue the /SWITCH OLDS command to force an IMS OLDS switch.
9. Quiesce and stop the asynchronous or Java drivers.

5.3 Post-measurement Procedure

After completing the measurement, the following general steps were used to capture the performance data:

1. Run the SMF logstream dump utility (IFASMFDL) to allow for post processing.
2. Run RMF post-processing against the dumped SMF data to produce various RMF reports detailing z/OS system activity.
3. Run IMS Performance Analyzer (IMSPA) against all the IMS OLDS processed between step 4 and 8 of Section 5.2 to produce various reports detailing IMS activity.

5.4 Measurement Metrics and Analysis

The results of each performance evaluation include many different data points from:

- RMF providing information about z/OS and hardware resources such as CPU utilization, memory consumption, and I/O rates
- IMSPA providing IMS internal statistics such as transaction rate, logging rate, and latch contention rates

All this data is captured and saved for future research and analysis.

There are some basic metrics that apply to almost all workload measurements as shown in Table 16.

Table 16: Performance Metrics for IBM zSystems Processor Comparison

Metric	Description
CPU % Busy	The average percent busy across all general CPs on an LPAR during the measurement interval.
Total LPAR % Busy	The average percent busy across all general CPs on an LPAR during the measurement interval.
ETR (Tran/Sec)	External Transaction Rate is the observed average transaction rate in transactions per second (TPS) over the measurement interval captured from the RMF report.
ITR	Internal Transaction Rate is a projection of the observed transaction rate (ETR) to what the transaction rate would be if the processors were running at 100% CPU busy, assuming linear scaling.

	ITR was calculated by dividing ETR by CPU % Busy
Total IMS Response Time (msec)	The total IMS transaction response time from the IMSPA report.
Total General CPU µsec/Tran	The total CPU microseconds spent per transaction was calculated using the following formula: (Number of CPs * CPU % Busy * 1000000)/ETR
Total IMS CPU Service Time (sec) per transaction	The amount of CPU time consumed by the processors for all IMS address spaces captured from the RMF report. IMS address spaces include control region, DL/I region, DBRC region, SCI, OM and IRLM. It is reported on a per transaction basis by dividing the total CPU service time consumed during an interval by the number of transactions processed in that interval.
IMS Connect CPU Service Time (sec) per transaction	The amount of CPU time consumed by the processors for the IMS Connect address space captured from the RMF report. It is reported on a per transaction basis by dividing the total CPU service time consumed during an interval by the number of transactions processed in that interval.
Dependent Regions CPU Service Time (sec) per transaction	The amount of CPU time consumed by the processors for IMS dependent region address space captured from the RMF report. It is reported on a per transaction basis by dividing the total CPU service time consumed during an interval by the number of transactions processed in that interval.
z/OS Connect CPU Service Time (sec) per transaction	The amount of CPU time consumed by the processors for the z/OS Connect address space captured from the RMF report. It is reported on a per transaction basis by dividing the total CPU service time consumed during an interval by the number of transactions processed in that interval.
CQS CPU Service Time (sec) per transaction	The amount of CPU time consumed by the processors for the CQS address space, if applicable, captured from the RMF report. It is reported on a per transaction basis by dividing the total CPU service time consumed during an interval by the number of transactions processed in that interval.
IXLOGR CPU Service Time (sec) per transaction	The amount of CPU time consumed by the processors for the z/OS Logger (IXGLOGR) address space, if applicable, captured from the RMF report. It is reported on a per transaction basis by dividing the total CPU service time consumed during an interval by the number of transactions processed in that interval.

Number of Offloads	The number of z/OS logger offloads, if applicable, captured from the RMF report.
Avg. CSA Below 16MB Key 7	The average usage of Key 7 common storage below 16 MB from the RMF report.
Avg. CSA Above 16MB Key 7	The average usage of Key 7 common storage above 16 MB from the RMF report.
Avg. LSQA Private	The average usage of private LSQA storage below 16 MB from the RMF report.
Avg. LSQA EPrivate	The average usage of extended private LSQA storage above 16 MB from the RMF report.
Avg. USER Private	The average usage of private USER storage below 16 MB from the RMF report.
Avg. User EPrivate	The average usage of extended private USER storage above 16 MB from the RMF report.

The IMS shared queues metrics were calculated using both LPARs in a data sharing environment as shown in Table 17.

Table 17: Performance Metrics for Shared Message Queues Comparison

Combined Metric	Description
Avg. CPU % Busy	The average percent busy across two LPARs during the measurement interval.
Total ETR (Tran/Sec)	The sum of the average transaction rates in transactions per second (TPS) during the measurement interval captured from the RMF report on both LPARs.
Combined ITR	The combined ITR during measurement interval calculated by dividing Total ETR with Avg. CPU % Busy.
Average Total IMS Response Time (msec)	The average of the total IMS transaction response time from the IMSPA reports on both LPARs.
Combined General CPU µsec/Tran	The total CPU microseconds spent per transaction calculated using the following formula: (Sum of total number of CPs on both LPARs * Avg. CPU % Busy * 1000000)/Total ETR
IMS CPU Service Time (sec) per transaction	The sum of total CPU time consumed by the processors for all IMS address spaces captured from the RMF report on both LPARs. IMS address spaces include control region, DL/I region, DBRC region, SCI, OM, and IRLM. It is reported on a per transaction basis by dividing the sum of total CPU service time consumed during an interval by the sum of total number of transactions processed in that interval on both LPARs.

IMS Connect CPU Service Time (sec) per transaction	The sum of total CPU time consumed by the processors for the IMS Connect address spaces captured from the RMF report on both LPARs. It is reported on a per transaction basis by dividing the sum of total CPU service time consumed during an interval by the sum of total number of transactions processed in that interval on both LPARs.
Dependent Regions CPU Service Time (sec) per transaction	The sum of total CPU time consumed by the processors for IMS dependent region address space captured from the RMF report on both LPARs. It is reported on a per transaction basis by dividing the sum of total CPU service time consumed during an interval by the sum of total number of transactions processed in that interval on both LPARs.
CQS CPU Service Time (sec) per transaction	The sum of total CPU time consumed by the processors for the CQS address space captured from the RMF report on both LPARs. It is reported on a per transaction basis by dividing the sum of total CPU service time consumed during an interval by the sum of total number of transactions processed in that interval on both LPARs.
IXGLOGR CPU Service Time (sec) per transaction	The sum of total CPU time consumed by the processors for the z/OS Logger (IXGLOGR) address space captured from the RMF report on both LPARs. It is reported on a per transaction basis by dividing the sum of total CPU service time consumed during an interval by the sum of total number of transactions processed in that interval on both LPARs.
Total Number of Offloads	The total number of z/OS logger offloads on both LPARs.

ITR is one way of comparing the processor efficiency of two IBM zSystems processors using the same software environment. ITR normalizes the observed transaction rate to the engine capacity of the machine. It essentially answers the question "assuming the transaction rate scales linearly with CPU usage, what is the maximum transaction rate possible on this particular hardware configuration (that is, when CPU percent busy is 100%)".

Note that ITR does not take into consideration other possible bottlenecks besides CPU (for instance, I/O, latch contention, or lock contention) that could further limit the theoretical maximum transaction rate.

When two different zSystems processors running the same software configuration are compared, IMS workload, and running at similar total CPU utilization, the processor with a larger ITR value is generally more efficient in terms of CPU consumed per transaction. Comparisons are considered equivalent in this document where the ITR difference is within +/-2%.

6 z16 Performance

6.1 Introduction

The IBM zSystems platform has been evolving with enhancements to support the demand for smarter solutions that leverage large volumes of data and deliver 24x7 availability, scalability, and security.

The z16 improved on its predecessor further with even higher capacity, processing power, resiliency, and data protection without any application change while maintaining core workload strategies of data serving and transaction processing.

This section describes the performance evaluations comparing z16 against z15 using the following IMS workloads:

- Full function with HALDB
- Data sharing full function with HALDB and shared message queues
- Fast Path banking
- CICS IMS DBCTL
- IMS TM-Db2 IRWW
- z/OS Connect with IMS SP
- Java message processing
- Open database management

The ITR values for each of the specific types of IMS workloads were used to evaluate the general performance and CPU efficiency of the new z16 against z15.

The software configuration was kept constant for any given pair of z16 and z15 measurement comparisons. Additionally, other than the processor type (z15 versus z16), the hardware configuration for example, the number of processors, I/O channels, DASD, and LPAR memory were also kept constant for each pair of z16 and z15 measurements.

All the workload evaluations were executed on the following machine and environment configurations as shown in Table 18.

Table 18: Performance Evaluation Environment

Hardware and Software Environment	
Processor	IBM z15 Model 8561-722 (T01) or IBM z16 Model 3931-7A0 (A01)
DASD	IBM System Storage DS8900F
IBM z/OS Operating System	z/OS Version 2 Release 5
IBM Enterprise COBOL for z/OS	Version 6 Release 3
IMS	IMS Version 15.3
z/OS Resource Measurement Facility (RMF)	Version 2 Release 5
IMS Performance Analyzer for z/OS (IMSPA)	Version 4 Release 4
z/OS Resource Access Control Facility (RACF®)	Version 2 Release 4
Java	Java 8 Service Release 7

6.2 Full-Function (FF) with High Availability Large Database (HALDB) Performance Evaluation

Full-function databases support the full set of IMS database functions and can be used in a variety of IMS applications.

This evaluation uses the FF with HALDB workload as described in Section 4.1. The workload consists of a mix of OSAM and VSAM with HDAM, HIDAM, PHDAM, and PHIDAM databases using inventory, hotel, and warehouse-type transactions that perform read, replace, delete, and insert database calls.

The objective of the FF with HALDB evaluation was to compare the ITR between z16 and z15 using the same software configuration with TCP/IP message protocols.

6.2.1 System Configuration

The FF with HALDB evaluation was executed on both z16 and z15 configured in a two-LPAR configuration as shown in Figure 5:

- LPAR 1 hosts IMS, 256 MPP regions, and IMS Connect with four general purpose engines

- LPAR 2 hosts asynchronous driver driving a up to 300 IMS Connect clients via TCP/IP with four general purpose engines

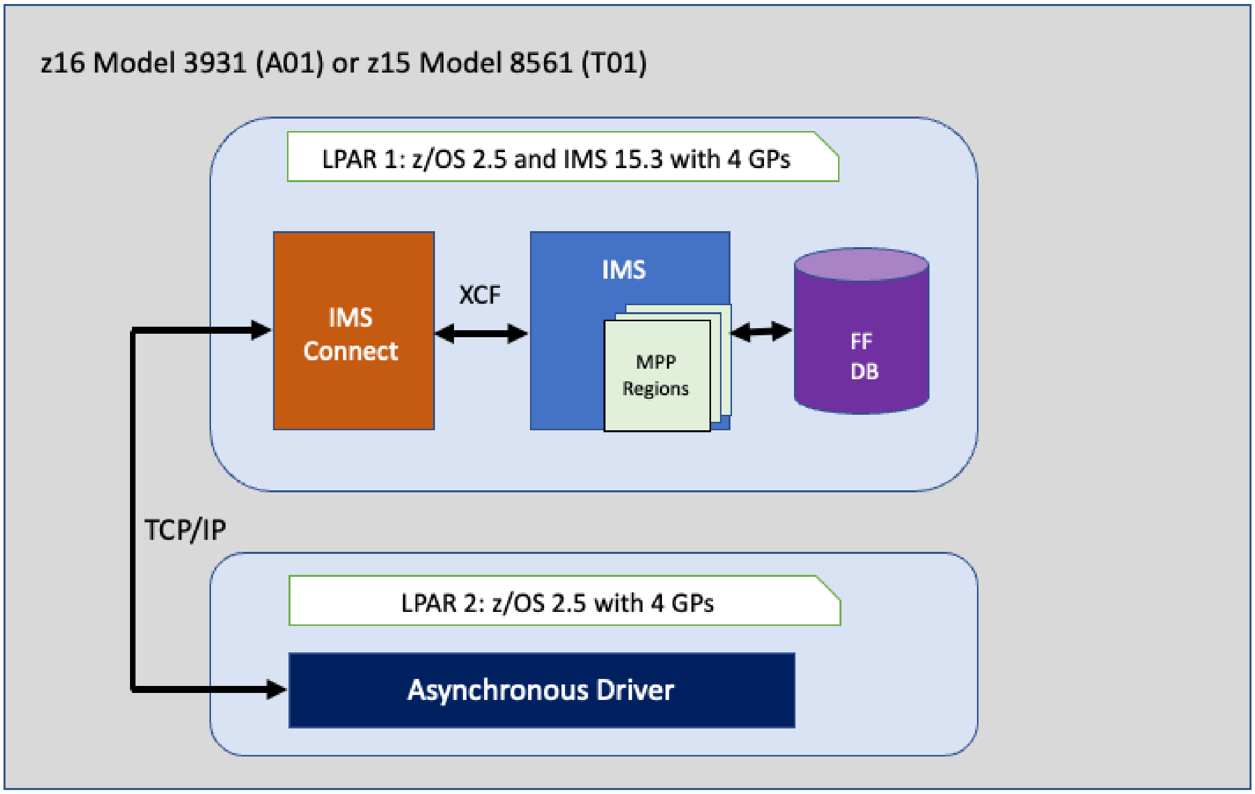


Figure 5: Full-Function with HALDB Environment Configuration

6.2.2 Evaluation Results

The z16 showed an improvement in ETR and ITR at similar CPU usage over z15 for the FF with HALDB workload. Table 19 shows the comparison between z16 and z15.

Table 19: Full-Function with HALDB Evaluation Results

Full-Function with HALDB Workload Evaluation				
	IBM z15	IBM z16	Delta	Delta %
CPU % Busy	80.21%	79.01%	-1.20%	-1.50%
ETR (Tran/Sec)	8229.65	9432.78	1203.13	14.62%
ITR	10260.13	11938.72	1678.59	16.36%

Total IMS Response Time (msec)	10.543	10.332	-0.211	-2.00%
Total General CPU µsec/Tran	389.859	335.044	-54.815	-14.06%
IMS CPU Service Time/Tran (µsec)	44.496	40.535	-3.961	-8.90%
ICON CPU Service Time/Tran (µsec)	17.777	14.831	-2.946	-16.57%
MPP CPU Service Time/Tran (µsec)	277.401	238.559	-38.842	-14.00%
Common Storage Below and Above 16 MB for Avg. Key 7				
Avg. CSA Below 16M Key 7 (K)	304	304	0	0.00%
Avg. CSA Above 16M Key 7 (M)	23.1	23.2	0.1	0.43%
Private Storage IMS Control Region				
Avg. LSQA Private (K)	604	604	0	0.00%
Avg. LSQA EPrivate (M)	15.1	15.1	0	0.00%
Avg. USER Private (K)	2276	2274	-2	-0.09%
Avg. USER EPrivate (M)	59.9	59.8	-0.1	-0.17%
Private Storage IMS DL/I Region				
Avg. LSQA Private (K)	1684	1688	4	0.24%
Avg. LSQA EPrivate (M)	13.3	13.3	0	0.00%
Avg. USER Private (K)	720	720	0	0.00%
Avg. USER EPrivate (M)	267	267	0	0.00%
Private Storage IMS Connect Region				
Avg. LSQA Private (K)	376	376	0	0.00%

Avg. LSQA EPrivate (M)	14	14	0	0.00%
Avg. USER Private (K)	56	56	0	0.00%
Avg. USER EPrivate (M)	181	181	0	0.00%

A series of scaling tests were run to compare z16 against z15 at various CPU percent busy values. Figure 6 shows the ITR versus transaction rate comparison, and Figure 7 shows the IMS response time versus transaction rate comparison. Figure 8 shows the CPU percent busy versus transaction rate comparison demonstrating lower CPU usage for z16 at various transaction rates.

The FF with HALDB workload on z16 showed the following improvements over IBM z15:

- Up to 16.36% ITR improvement for z16 as compared to z15
- Improved IMS transaction response time in z16
- Reduction in total IMS CPU service time per transaction in z16
- Lower CPU usage for z16 compared to z15

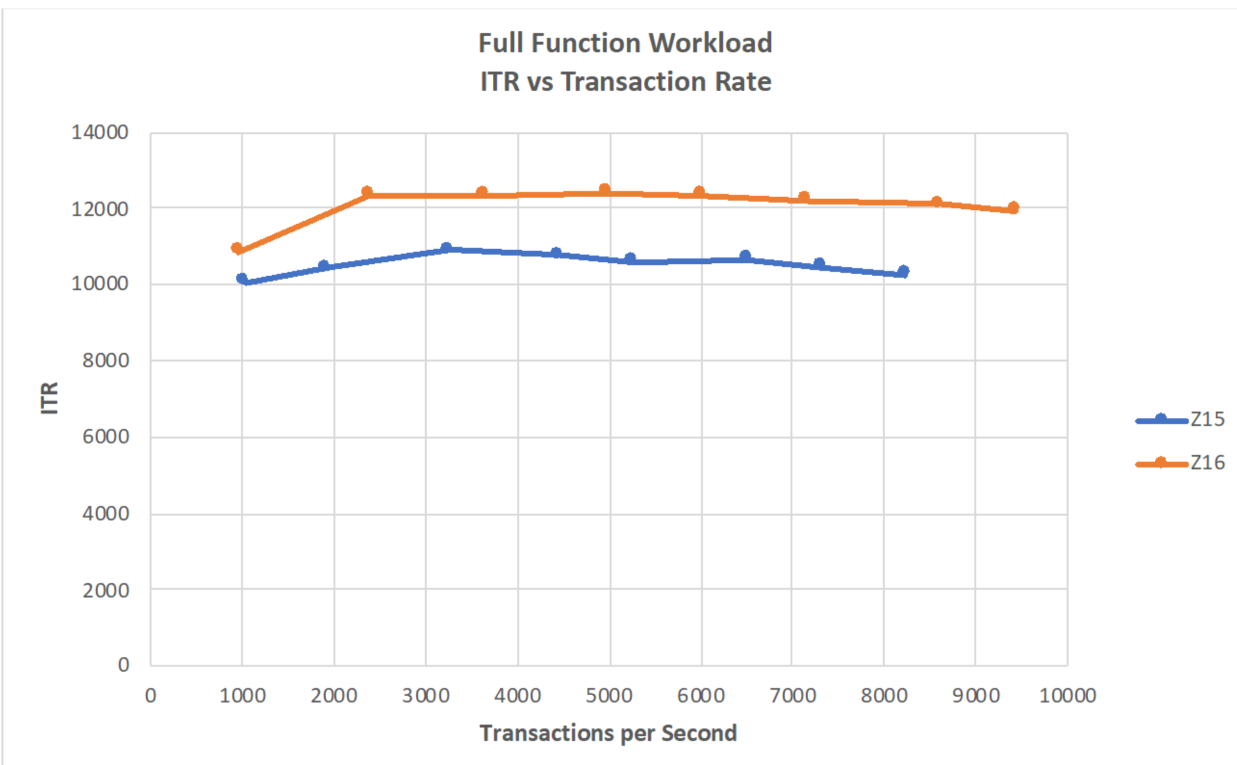


Figure 6: Full-Function with HALDB ITR versus Transaction Rate Comparison Results

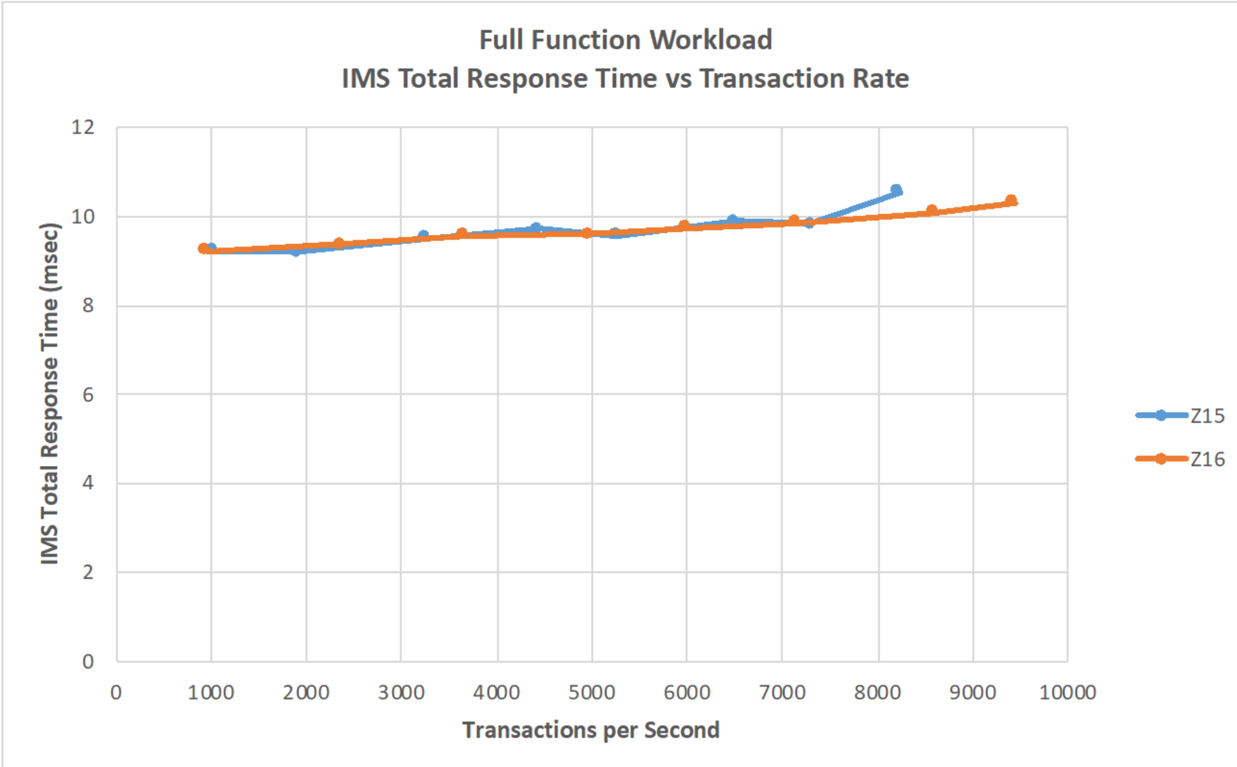


Figure 7: Full-Function with HALDB IMS Total Response Time versus Transaction Rate Comparison Results

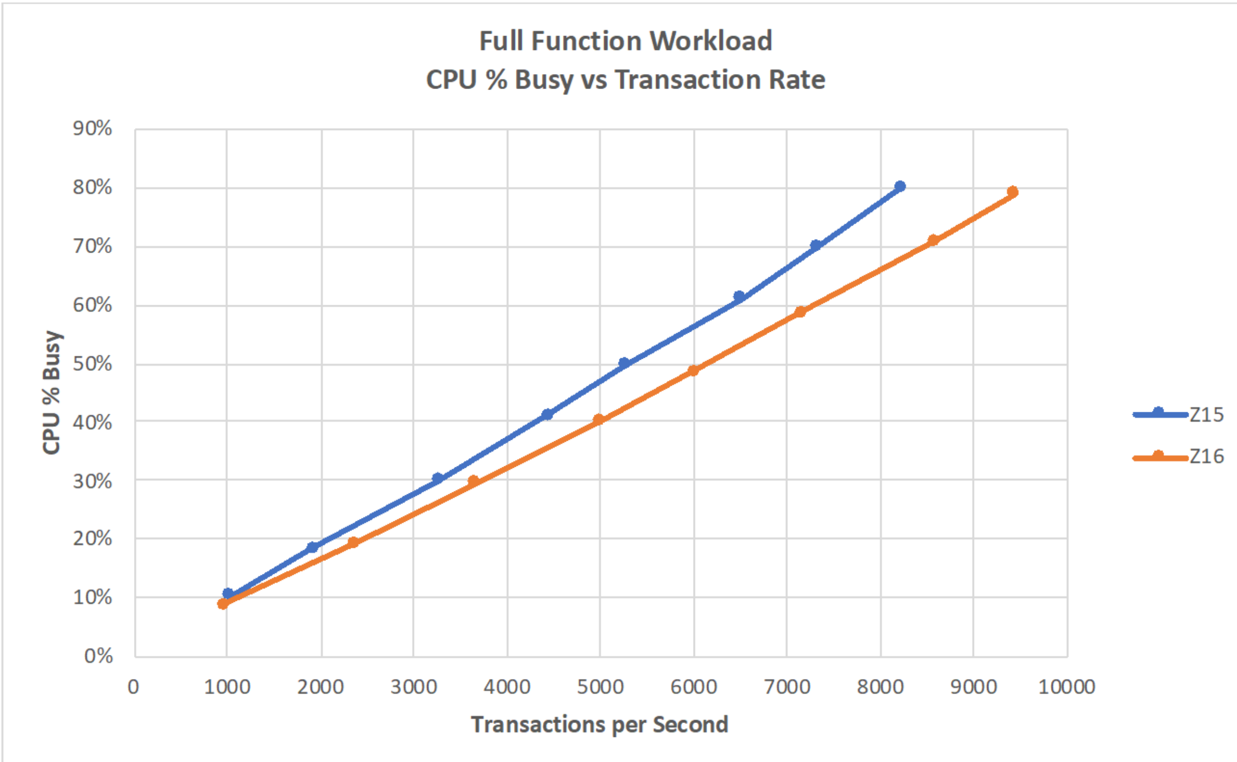


Figure 8: Full-Function with HALDB Total CPU % Busy versus Transaction Rate Comparison Results

6.3 Data Sharing Full-Function (DSFF) with High Availability Large Database and Shared Message Queue (SMQ) Performance Evaluation

The DSFF with HALDB and SMQ workload uses the CQS, which manages a shared queue of messages residing on a CF list structure in a data sharing environment.

This evaluation uses the FF with HALDB workload described in Section 4.2.

The objective of the DSFF with HALDB and SMQ evaluation was to compare the ITR between z16 and z15 in the same software configuration with TCP/IP message protocol.

6.3.1 System Configuration

The DSFF with HALDB and SMQ evaluation was executed on both z16 and z15 in a three-LPAR configuration as shown in Figure 9:

- LPAR 1 hosts IMS, 256 MPP regions, and IMS Connect with four general purpose engines

- LPAR 2 hosts IMS, 256 MPP regions, and IMS Connect with four general purpose engines
- LPAR 3 hosts asynchronous driver driving 300 IMS Connect clients into each IMS via TCP/IP with six general purpose engines

Each IMS is started with CQS and uses a shared message queue structure and a z/OS log stream structure for the CQS log residing in an ICF.

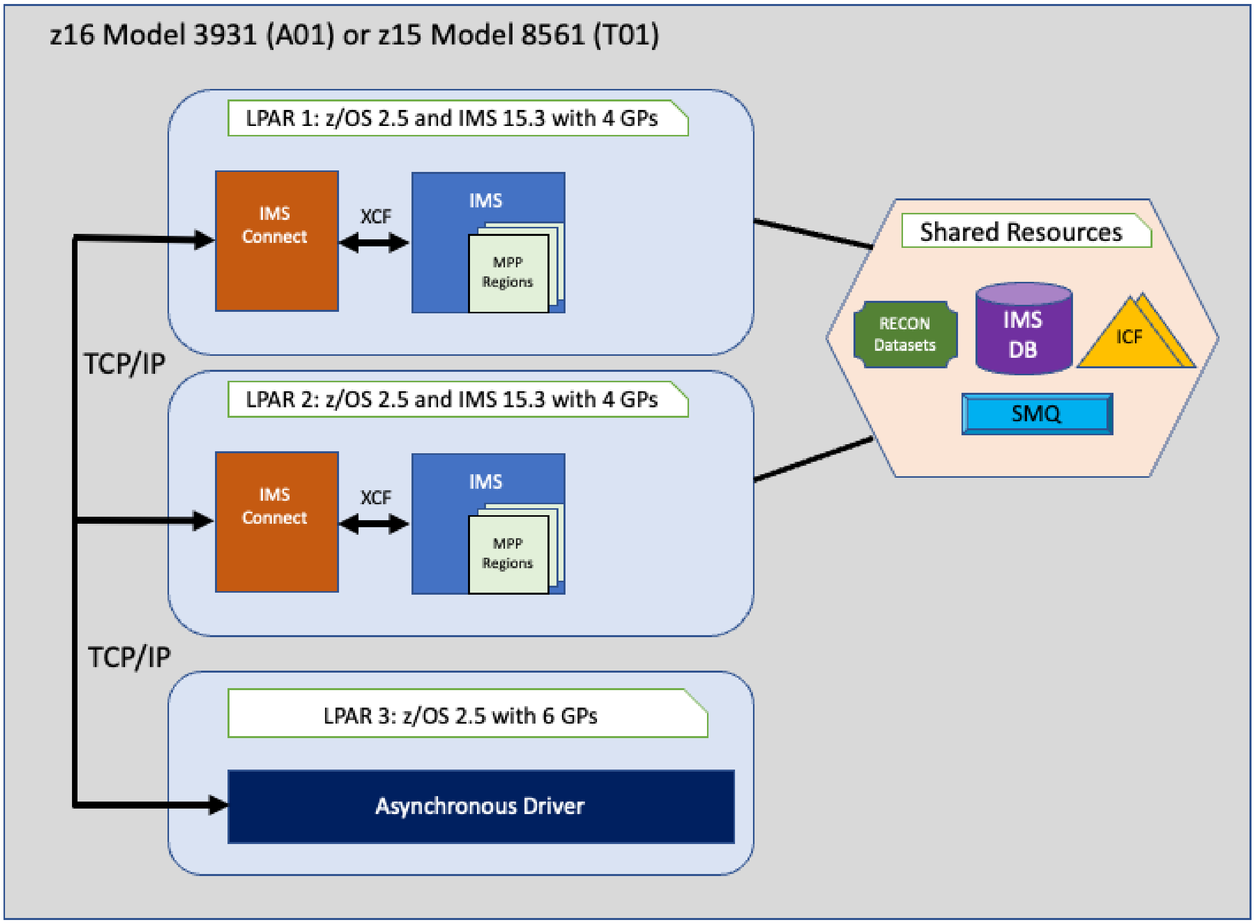


Figure 9: DSFF with HALDB and SMQ Environment Configuration

Evaluation Results

The z16 showed an improvement in ETR and ITR at similar CPU usage over z15 for the DSFF with HALDB and SMQ workload. Table 20 and Table 21 show the comparison between z16 and z15.

The data shown here for z15 and z16 combined is for both IMSs in the Shared Queues environment calculated using formulas from Table 17 in Section 5.4.

Table 20: DSFF with HALDB and SMQ Evaluation Results

DSFF with HALDB and SMQ Workload Evaluation								
	IBM z15		IBM z16					
	IMS1	IMS2	IMS1	IMS2	z15 Combined	z16 Combined	Delta	Delta %
CPU % Busy	80.33 %	79.60%	81.62%	74.96%	79.97%	78.29%	-1.68%	-2.09%
ETR (Tran/Sec)	4985.75	4995.38	5015.62	5137.49	9981.13	10153.11	171.98	1.72%
ITR	6206.59	6275.60	6145.09	6853.64	12482.19	12998.73	516.54	4.14%
Average Total IMS Response Time (msec)	13.180	13.231	13.312	13.218	13.206	13.265	0.059	0.45%
Total General CPU µsec/Tran	644.477	637.389	650.927	583.631	1281.866	1234.558	-47.308	-3.69%
IMS CPU Service Time/tran (µsec)	64.167	63.175	57.778	57.348	127.342	115.126	-12.216	-9.59%
ICON CPU Service Time/Tran (µsec)	17.810	18.367	15.218	14.698	36.177	29.916	-6.261	-17.31%
MPP CPU Service Time/Tran (µsec)	413.572	413.572	428.694	378.269	827.144	806.963	-20.181	-2.44%

CQS CPU Service Time/Tran (µsec)	48.109	48.127	63.097	47.017	96.236	110.114	13.878	14.42%
IXGLOGR CPU Service Time/Tran (µse)	6.629	7.177	5.922	6.751	13.806	12.673	-1.133	-8.21%
Number of offloads	1	1	0	1	2	1	-1	-50.00%

Table 21: DSFF with HALDB and SMQ Evaluation Storage Usage

Common Storage Below and Above 16MB for Avg. Key 7				
	IMS1		IMS2	
	z15	z16	z15	z16
Avg. CSA Below 16M Key 7 (K)	304	304	304	304
Avg. CSA Above 16M Key 7 (M)	27.7	27.8	27.7	27.8
Private Storage IMS Control Region				
Avg. LSQA Private (K)	622	620	623	620
Avg. LSQA EPrivate (M)	16	16	16	16
Avg. USER Private (K)	2302	2288	2330	2306
Avg. USER EPrivate (M)	60.7	60.5	61.2	60.4
Private Storage IMS DL/I Region				
Avg. LSQA Private (K)	1688	1676	1688	1708
Avg. LSQA EPrivate (M)	14.6	14.2	14.2	14.3
Avg. USER Private (K)	720	720	720	720
Avg. USER EPrivate (M)	268	268	268	268

Private Storage IMS Connect Region				
Avg. LSQA Private (K)	376	376	376	376
Avg. LSQA EPrivate (M)	14	14	14	14.2
Avg. USER Private (K)	56	56	56	56
Avg. USER EPrivate (M)	178	178	178	178
Private Storage CQS Region				
Avg. LSQA Private (K)	348	356	352	352
Avg. LSQA EPrivate (M)	14	13.6	13.8	13.7
Avg. USER Private (K)	16	16	16	16
Avg. USER EPrivate (M)	11.7	11.7	11.7	11.7

A series of scaling tests were run to compare z16 against z15 at various CPU percent busy values. Figure 10 shows the ITR versus transaction rate comparison, and Figure 11 shows the IMS response time versus transaction rate comparison. Figure 12 shows the CPU percent busy versus transaction rate demonstrating lower CPU usage for z16 at various transaction rates.

The DSFF with HALDB and SMQ workload on z16 showed the following improvements over z15:

- Up to 4% ITR improvement for z16 as compared to z15
- Improved IMS transaction response time in z16
- Reduction of almost 9.5% in total IMS CPU service time per transaction in z16
- Lower CPU usage for IBM z16 compared to z15

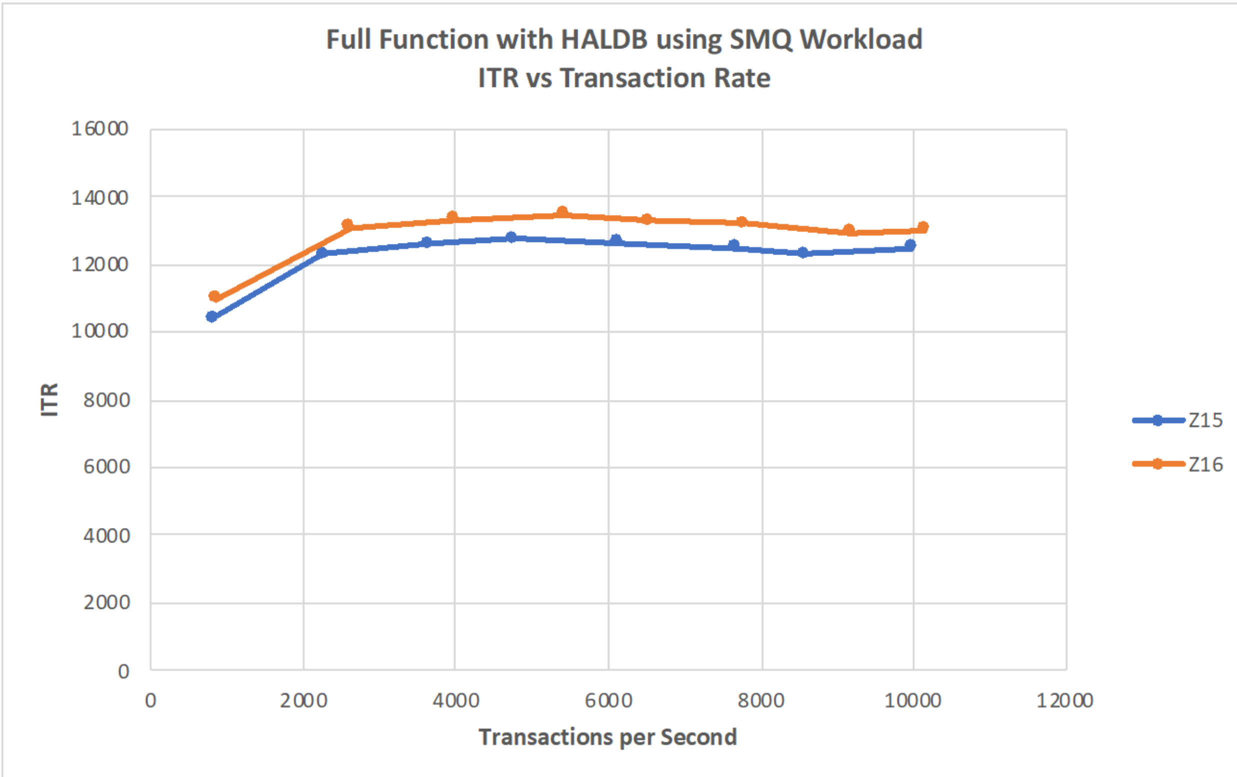


Figure 10: DSFF with HALDB and SMQ ITR versus Transaction Rate Comparison Results

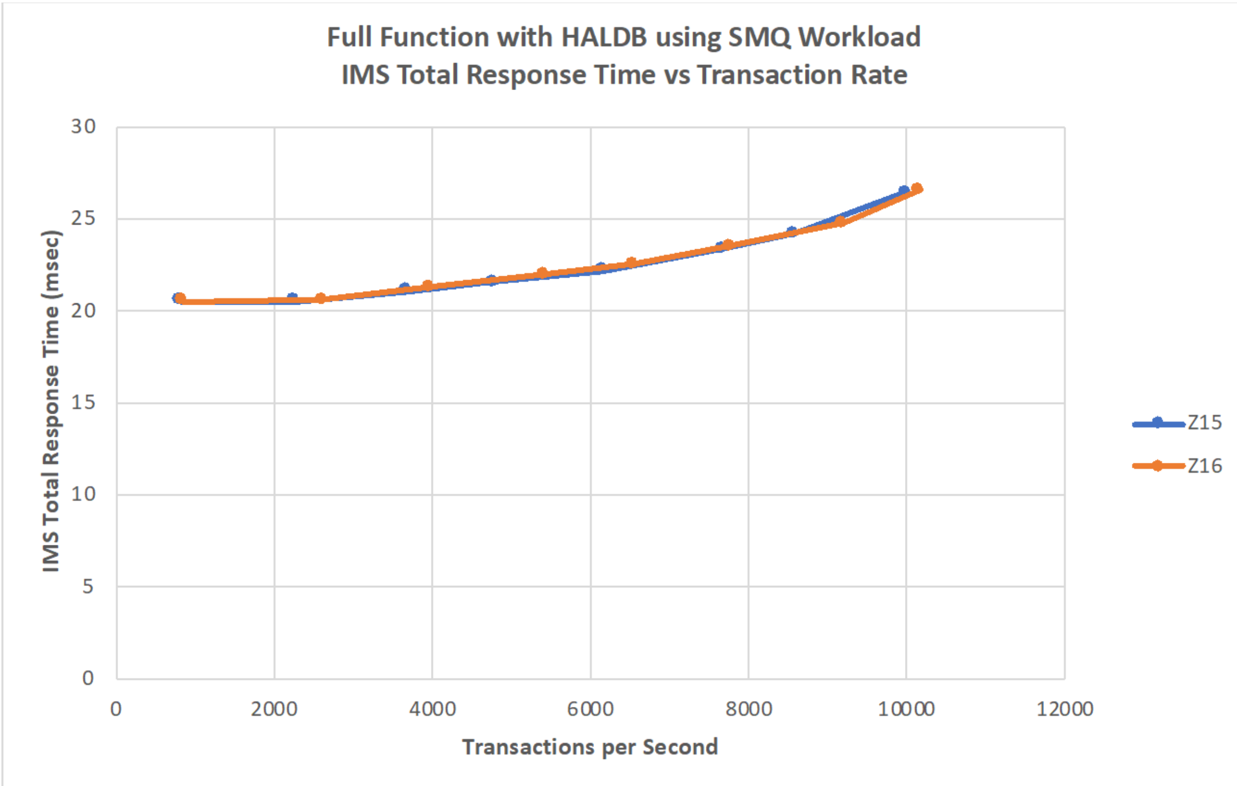


Figure 11: DSFF with HALDB and SMQ IMS Total Response Time versus Transaction Rate Comparison Results

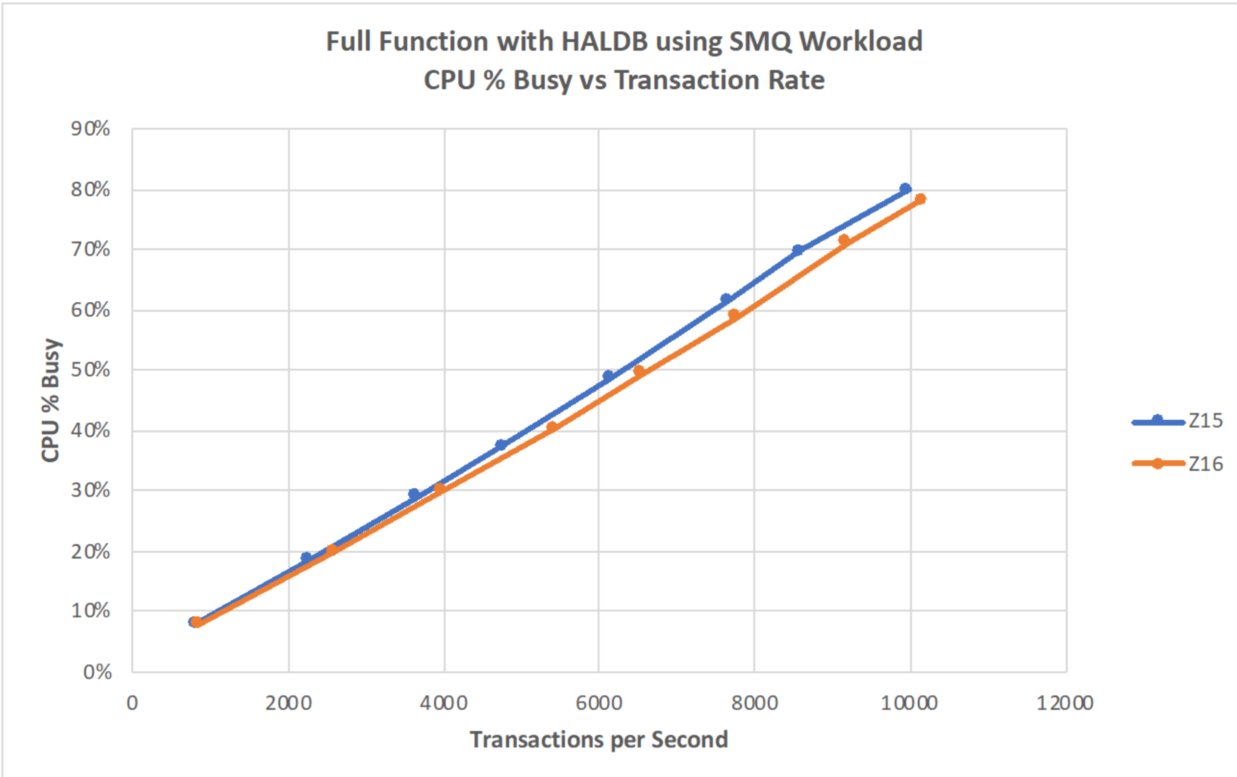


Figure 12: DSFF with HALDB and SMQ Total CPU % Busy versus Transaction Rate Comparison Results

6.4 Fast Path (FP) Banking Performance Evaluation

FP is capable of performing transaction and database processing at high rates. If your system requirements include a high transaction volume with relatively uncomplicated database structure and message processing (for example, no logical relationships, no message switching), FP can be advantageous over FF processing. The FP evaluation measured the following scenario:

- TCP/IP input through IMS Connect with FP 64-bit buffer manager enabled

The FP 64-bit buffer manager was introduced in IMS V11, and it autonomically controls the number and size of FP buffer pools for DEDBs. This autonomic control eliminates the need for customers to manually define FP buffer pools during system definition.

This evaluation used the FP banking workload, which includes credit card type transactions such as CCCK (Credit Card Check), CLCK (Credit Card Limit Check), CREDIT, DEBIT, and LOST card reporting as described in Section 4.3.

The objective of the FP evaluation was to compare the ITR between z16 and z15 in the same software environment with TCP/IP message protocol.

6.4.1 System Configuration

The FP evaluation was executed on both z16 and z15 in a two-LPAR configuration as shown in Figure 13:

- LPAR 1 hosts IMS, 100 IFP regions, and two IMS Connect with six general purpose engines
- LPAR 2 hosts two asynchronous drivers driving a total of 300 IMS Connect clients via TCP/IP with six general purpose engines

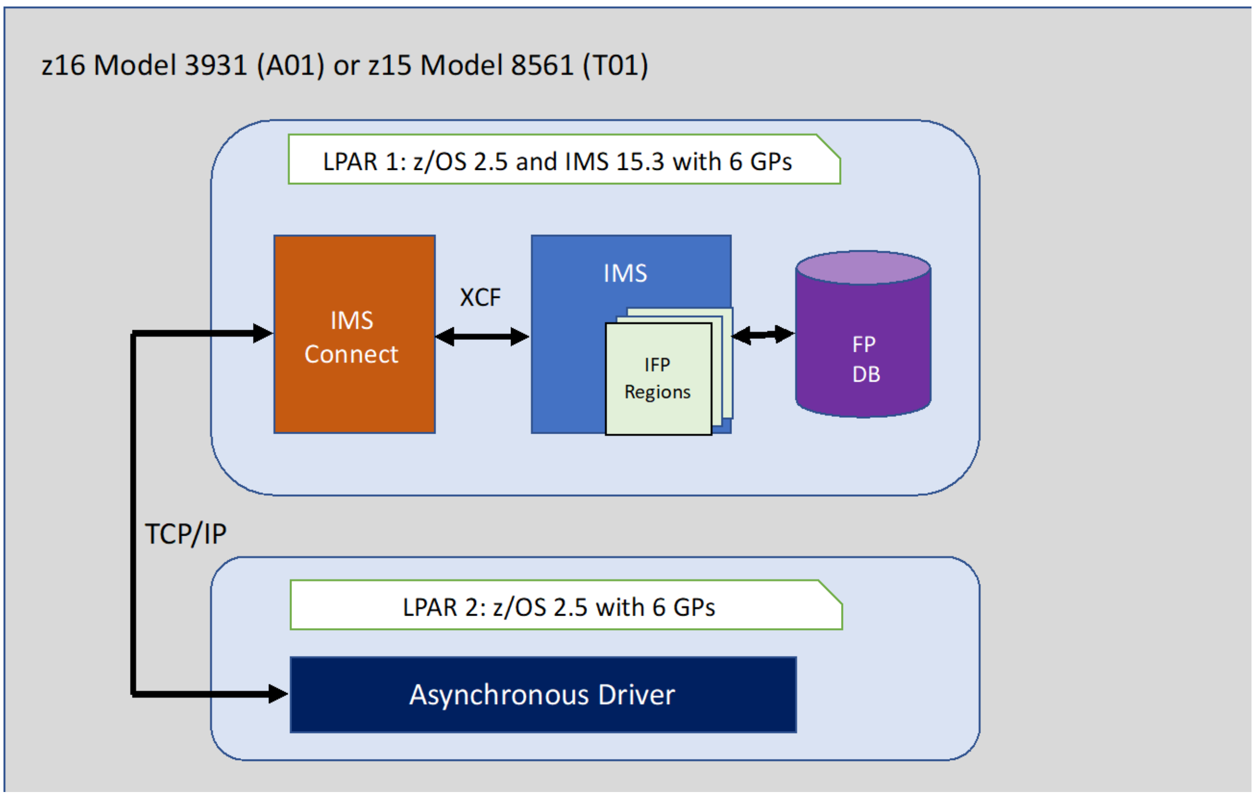


Figure 13: Fast Path Banking Environment Configuration

6.4.2 Evaluation Results

The z16 showed an improvement in ETR and ITR at similar CPU usage over z15 for the FP banking workload. Table 22 shows the results of the FP evaluation for both z16 and z15.

Table 22: Fast Path Evaluation Results

Fast Path Workload Evaluation				
	IBM z15	IBM z16	Delta	Delta %
CPU % Busy	79.78%	80.03%	0.25%	0.31%
ETR (Tran/Sec)	93588.61	104144.2	10555.59	11.28%
ITR	117308.36	130131.45	12823.09	10.93%
Total IMS Response Time (msec)	0.273	0.241	-0.032	-11.72%
Total General CPU µsec/Tran	51.147	46.107	-5.040	-9.85%
IMS CPU Service Time/Tran (µsec)	12.072	10.881	-1.191	-9.87%
ICON CPU Service Time/Tran (µsec)	13.332	12.447	-0.885	-6.64%
IFP CPU Service Time/Tran (µsec)	15.628	14.244	-1.384	-8.86%
Common Storage Below and Above 16 MB for Avg. Key 7				
Avg. CSA Below 16M Key 7 (K)	280	280	0	0.00%
Avg. CSA Above 16M Key 7 (M)	701	701	0	0.00%
Private Storage IMS Control Region				
Avg. LSQA Private (K)	1124	1120	-4	-0.36%
Avg. LSQA EPrivate (M)	14.9	14.8	-0.1	-0.67%
Avg. USER Private (K)	2272	2272	0	0.00%
Avg. USER EPrivate (M)	68.1	68.1	0	0.00%
Private Storage IMS DL/I Region				
Avg. LSQA Private (K)	360	356	-4	-1.11%

Avg. LSQA EPrivate (M)	9.28	9.27	-0.011	-0.12%
Avg. USER Private (K)	636	636	0	0.00%
Avg. USER EPrivate (M)	265	265	0	0.00%
Private Storage IMS Connect Region				
Avg. LSQA Private (K)	376	376	0	0.00%
Avg. LSQA EPrivate (M)	14.60	14.85	0.25	1.71%
Avg. USER Private (K)	56	56	0	0.00%
Avg. USER EPrivate (M)	178.5	179.0	0.5	0.28%

A series of scaling tests were run to compare z16 against z15 at various CPU percent busy values. Figure 14 shows the ITR versus transaction rate comparison, and Figure 15 shows the IMS response time versus transaction rate comparison. Figure 16 shows the CPU percent busy versus transaction rate demonstrating lower CPU usage for z16 at various transaction rates.

The FP banking workload on z16 showed the following improvements over z15:

- Up to 10.9% ITR improvement for IBM z16 as compared to z15
- Improved IMS transaction response time of 11.7% in z16
- Reduction of almost 10% in total IMS CPU service time per transaction in z16
- Lower CPU usage for z16 compared to z15

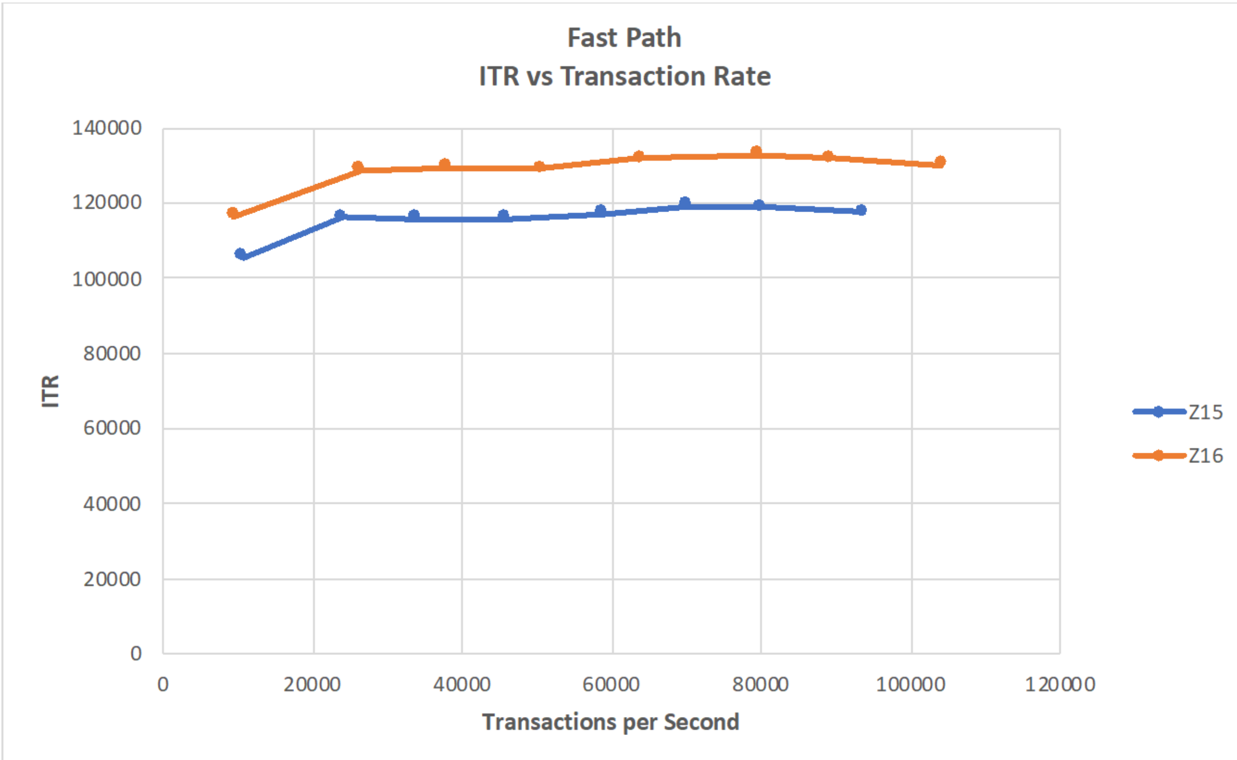


Figure 14: Fast Path Banking ITR versus Transaction Rate Comparison Results

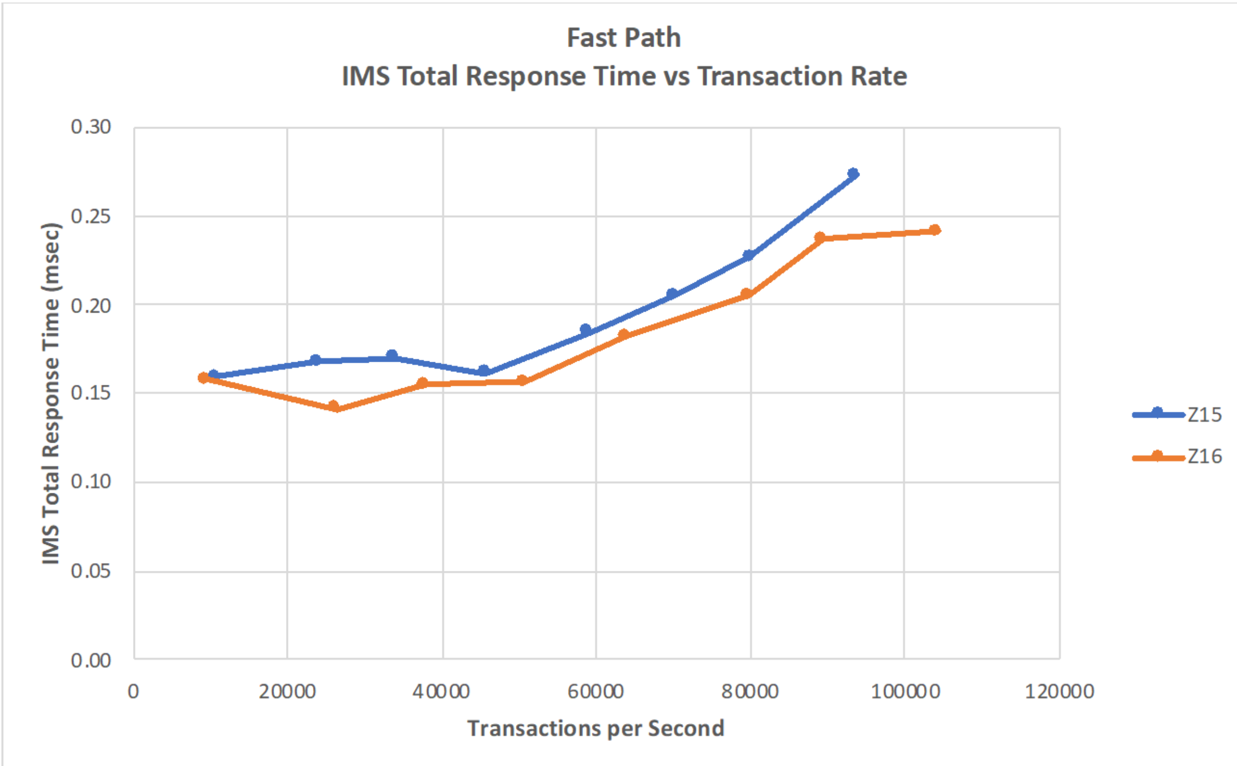


Figure 15: Fast Path Banking IMS Total Response Time versus Transaction Rate Comparison Results

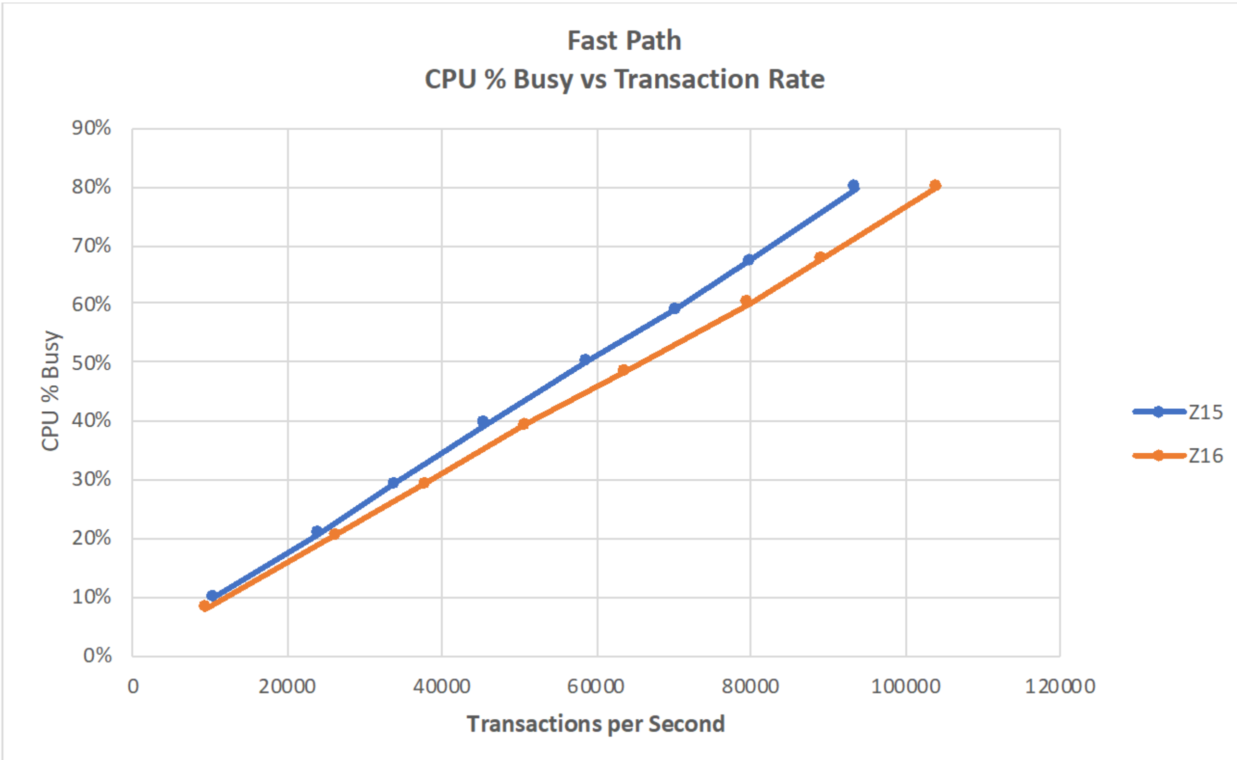


Figure 16: Fast Path Banking CPU % Busy versus Transaction Rate Comparison Results

6.5 CICS - IMS Database Control Performance Evaluation

The CICS IMS DBCTL evaluation was done for two workloads, IMS Full Function and IMS Fast Path, as described in Section 4.5. CICS Version 5 Release 6 was used with IMS enabled for Open Transaction Environment (OTE). The CICS program was driven by API interface using the z/OS Connect server services.

The objective of the CICS IMS DBCTL evaluation was to compare the ITR between z16 and z15 in the same software configuration using both full-function and Fast Path workloads.

6.5.1 System Configuration – CICS - IMS Database Control Full Function (FF)

The CICS IMS DBCTL FF evaluation was executed in both z16 and z15 in a two-LPAR configuration as shown in Figure 17:

- LPAR 1 hosts CICS and IMS using four general purpose engines
- LPAR 2 hosts asynchronous driver driving a total of 60 clients TCPIP with four general purpose engines

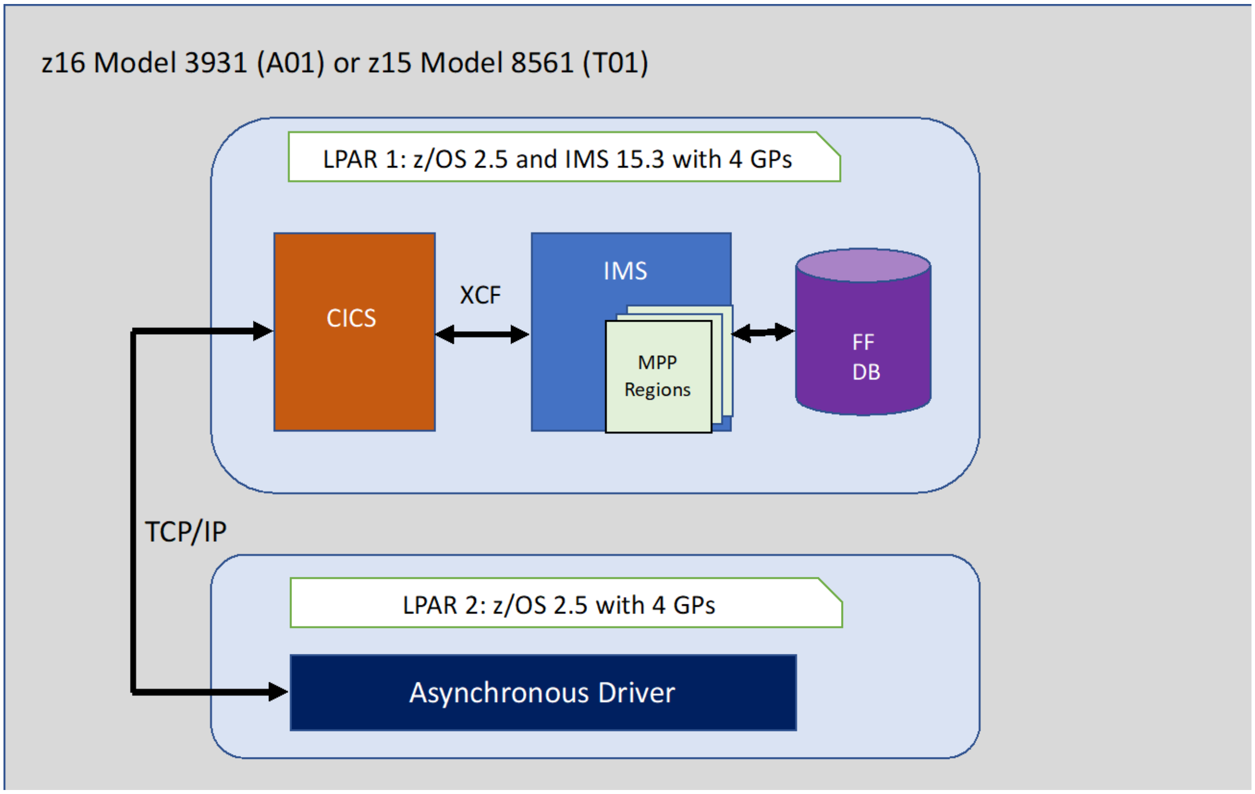


Figure 17: CICS IMS DBCTL Full-Function Environment Configuration

6.5.2 Evaluation Results – CICS - IMS Database Control Full Function (FF)

The z16 showed an improvement in ETR and ITR at similar CPU usage over z15 for the CICS IMS DBCTL FF workload. Table 23 shows the comparison between z16 and z15.

Table 23: CICS IMS DBCTL Full Function Evaluation Results

CICS IMS DBCTL Full Function Workload Evaluation				
	IBM z15	IBM z16	Delta	Delta %
CPU % Busy	80.43%	79.32%	-1.11%	-1.38%
ETR (Tran/Sec)	6507.80	7786.03	1278.23	19.64%
ITR	8091.26	9815.97	1724.71	21.32%

Total IMS Response Time (msec)	2.259	2.046	-0.213	-9.43%
Total General CPU µsec/Tran	494.361	407.499	-86.862	-17.57%
IMS CPU Service Time/Tran (µsec)	21.290	18.001	-3.289	-15.45%
CICS CPU Service Time/Tran (µsec)	281.388	223.452	-57.936	-20.59%
Common Storage Below and Above 16 MB for Avg. Key 7				
Avg. CSA Below 16M Key 7 (K)	236	236	0	0.00%
Avg. CSA Above 16M Key 7 (M)	15.5	15.7	0.2	1.29%
Private Storage IMS Control Region				
Avg. LSQA Private (K)	604	604	0	0.00%
Avg. LSQA EPrivate (M)	14.6	14.7	0.1	0.68%
Avg. USER Private (K)	2276	2276	0	0.00%
Avg. USER EPrivate (M)	54.1	54.1	0	0.00%
Private Storage IMS DL/I Region				
Avg. LSQA Private (K)	1372	1372	0	0.00%
Avg. LSQA EPrivate (M)	12.4	12.3	-0.1	-0.81%
Avg. USER Private (K)	704	704	0	0.00%
Avg. USER EPrivate (M)	266	266	0	0.00%
Private Storage CICS Region				
Avg. LSQA Private (K)	540	548	8	1.48%
Avg. LSQA EPrivate (M)	12.7	13.2	0.5	3.94%

Avg. USER Private (K)	4418	4428	10	0.23%
Avg. USER EPrivate (M)	125	125	0	0.00%

A series of scaling tests were run to compare z16 against z15 at various CPU percent busy values. Figure 18 shows the ITR versus transaction rate comparison, and Figure 19 shows the IMS response time versus transaction rate comparison. Figure 20 shows the CPU percent busy versus transaction rate demonstrating lower CPU usage for the z16 at various transaction rates. The CICS IMS DBCTL FF workload on z16 showed the following improvements over z15:

- Up to 21% ITR improvement for z16 as compared to z15
- Improved IMS transaction response time of about 9% in z16
- Reduction of about 15.5% in total IMS CPU service time per transaction in z16
- Lower CPU usage for z16 compared to z15

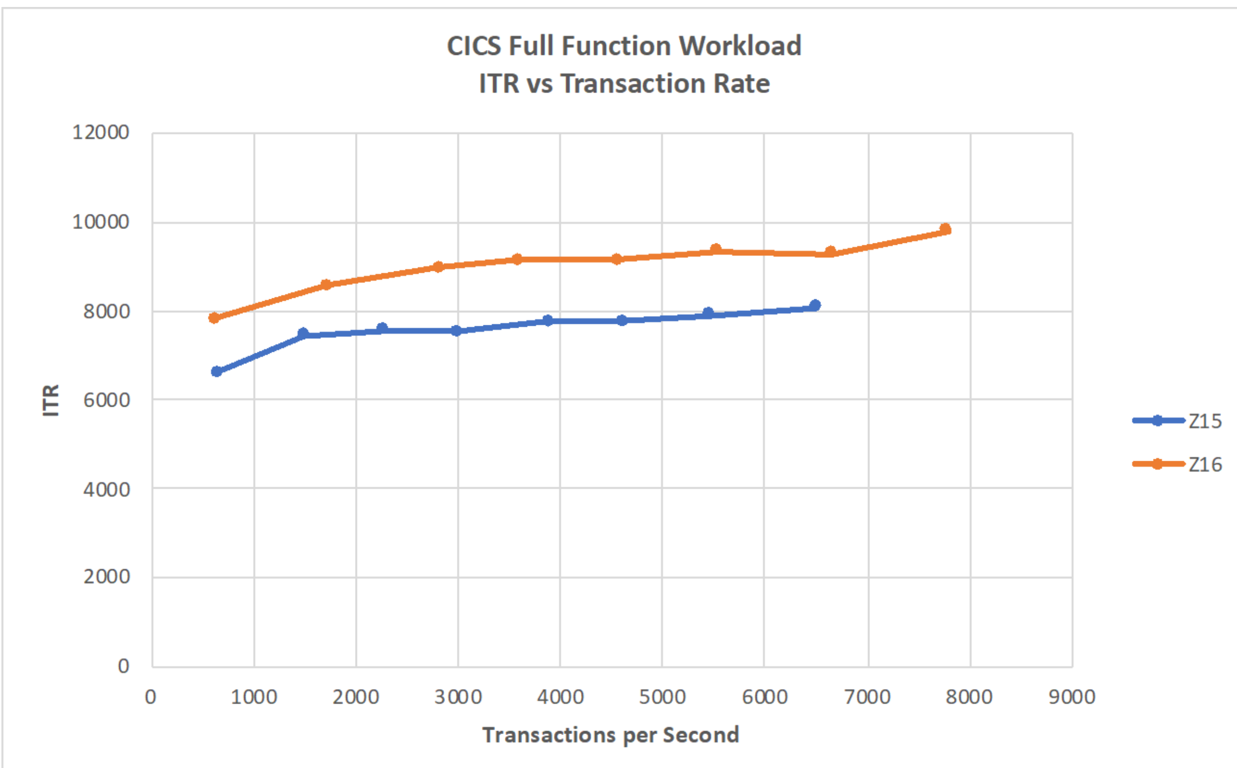


Figure 18: CICS IMS DBCTL Full Function ITR versus Transaction Rate Comparison Results

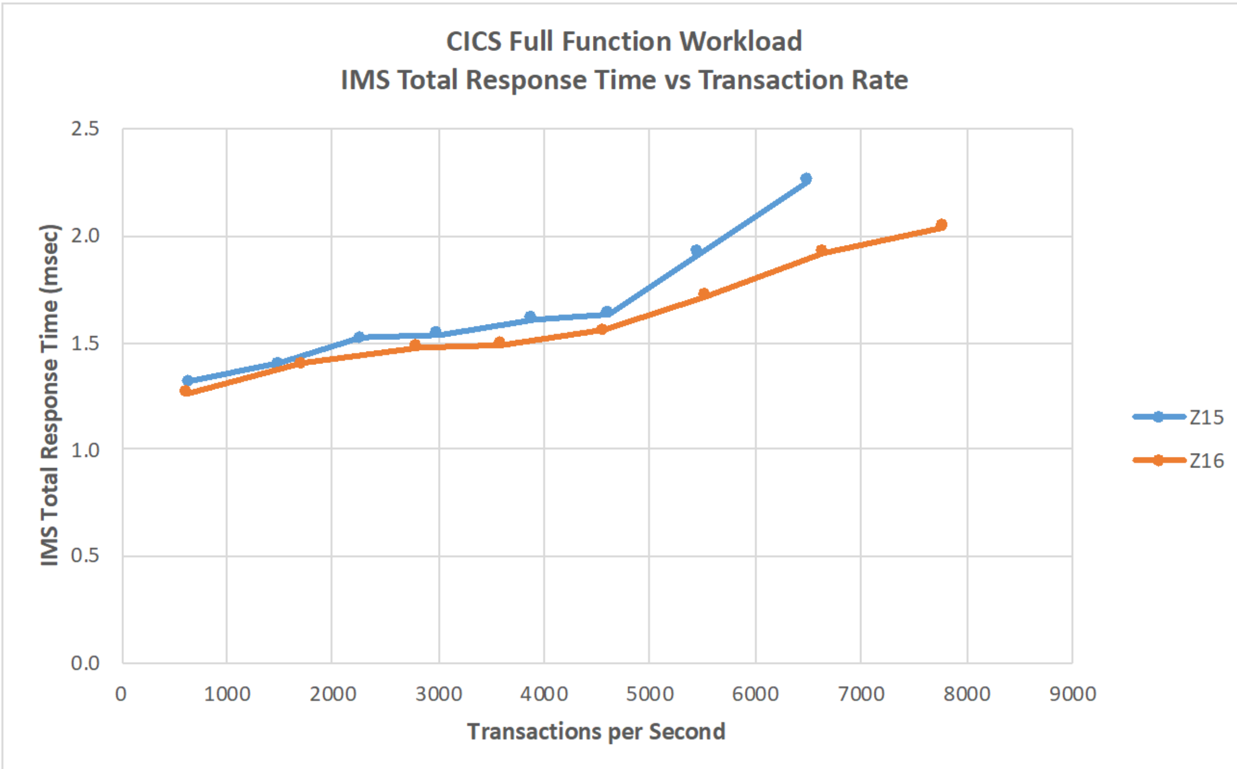


Figure 19: CICS IMS DBCTL Full Function IMS Total Response Time versus Transaction Rate Comparison Results

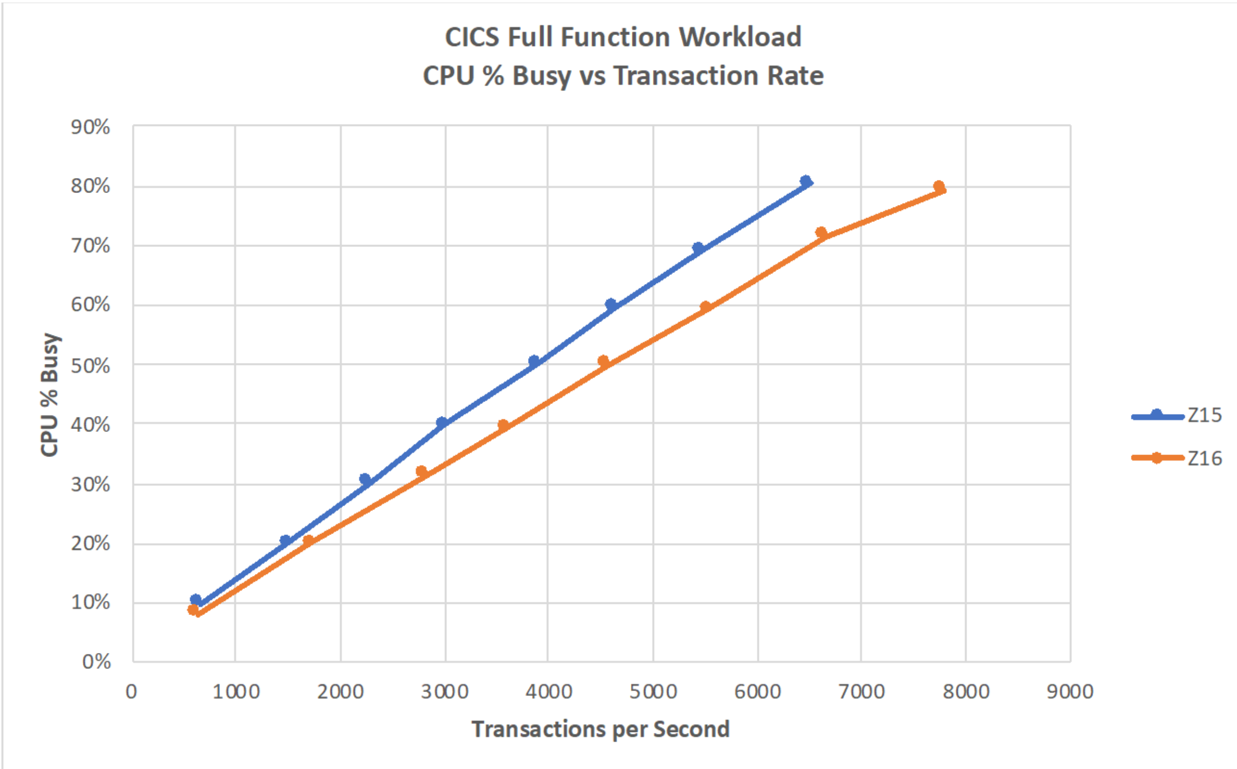


Figure 20: CICS IMS DBCTL Full Function CPU % Busy versus Transaction Rate Comparison Results

6.5.3 System Configuration – CICS - IMS Database Control Fast Path (FP)

The CICS IMS DBCTL FP evaluation was executed on both z16 and z15 in a two-LPAR configuration as shown in Figure 21:

- LPAR 1 hosts CICS and IMS using six general purpose engines
- LPAR 2 asynchronous driver driving a total of 45 clients with four general purpose engines

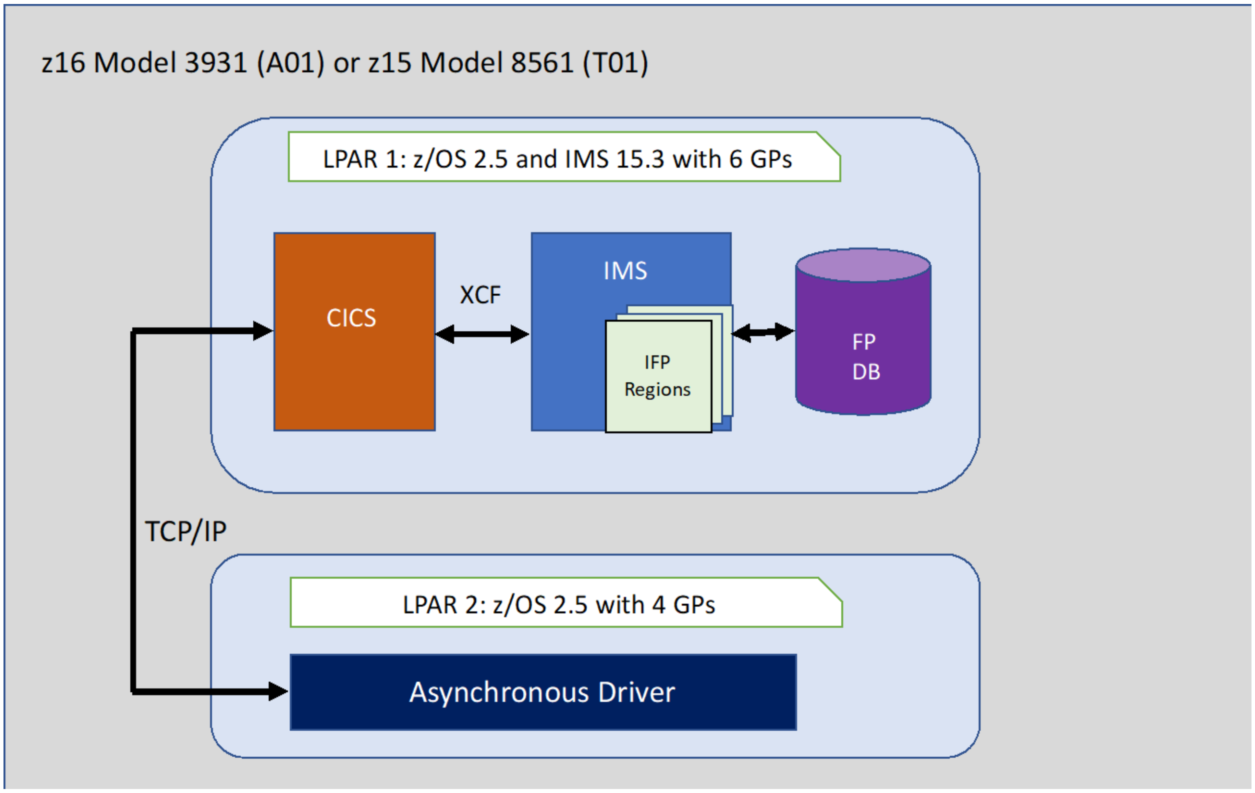


Figure 21: CICS IMS DBCTL Fast Path Environment Configuration

6.5.4 Evaluation Results – CICS - IMS Database Control Fast Path (FP)

The z16 showed an improvement in ETR and ITR at similar CPU usage over z15 for the CICS IMS DBCTL FP workload. Table 24 shows the comparison between z16 and z15.

Table 24: CICS IMS DBCTL Fast Path Evaluation Results

CICS IMS DBCTL Fast Path Workload Evaluation				
	IBM z15	IBM z16	Delta	Delta %
CPU % Busy	76.83%	77.57%	0.74%	0.96%
ETR (Tran/Sec)	14921.60	17618.82	2697.22	18.08%
ITR	19421.58	22713.45	3291.87	16.95%

Total IMS Response Time (msec)	0.417	0.417	0	0.00%
Total General CPU µsec/Tran	308.935	264.161	-44.774	-14.49%
IMS CPU Service Time/Tran (µsec)	19.516	16.882	-2.634	-13.50%
CICS CPU Service Time/Tran (µsec)	122.710	97.335	-25.375	-20.68%
Common Storage Below and Above 16 MB for Avg. Key 7				
Avg. CSA Below 16M Key 7 (K)	243	243	0	0.00%
Avg. CSA Above 16M Key 7 (M)	30.1	30.3	0.2	0.66%
Private Storage IMS Control Region				
Avg. LSQA Private (K)	1104	1104	0	0.00%
Avg. LSQA EPrivate (M)	14.1	14.2	0.1	0.71%
Avg. USER Private (K)	2276	2272	-4	-0.18%
Avg. USER EPrivate (M)	64.3	64.3	0	0.00%
Private Storage IMS DL/I Region				
Avg. LSQA Private (K)	344	340	-4	-1.16%
Avg. LSQA EPrivate (M)	9.29	9.26	-0.03	-0.32%
Avg. USER Private (K)	636	636	0	0.00%
Avg. USER EPrivate (M)	264	264	0	0.00%
Private Storage CICS Region				
Avg. LSQA Private (K)	528	534	6	1.14%
Avg. LSQA EPrivate (M)	12.5	12.3	-0.2	-1.60%
Avg. USER Private (K)	4402	4404	2	0.05%

Avg. USER EPrivate (M)	125	125	0	0.00%

A series of scaling tests were run to compare z16 against z15 at various CPU percent busy values. Figure 22 shows the ITR versus transaction rate comparison, and Figure 23 shows the IMS response time versus transaction rate comparison. Figure 24 shows the CPU percent busy versus transaction rate demonstrating lower CPU cost for the z16 at various transaction rates.

The CICS IMS DBCTL FP workload on z16 showed the following improvements over z15:

- Up to 16.9% ITR improvement for z16 as compared to z15
- Reduction of 13.5% in total IMS CPU service time per transaction in z16
- Improved Total General CPU service time per transaction of almost 14.4% in z16
- Lower CPU usage for z16 compared to z15

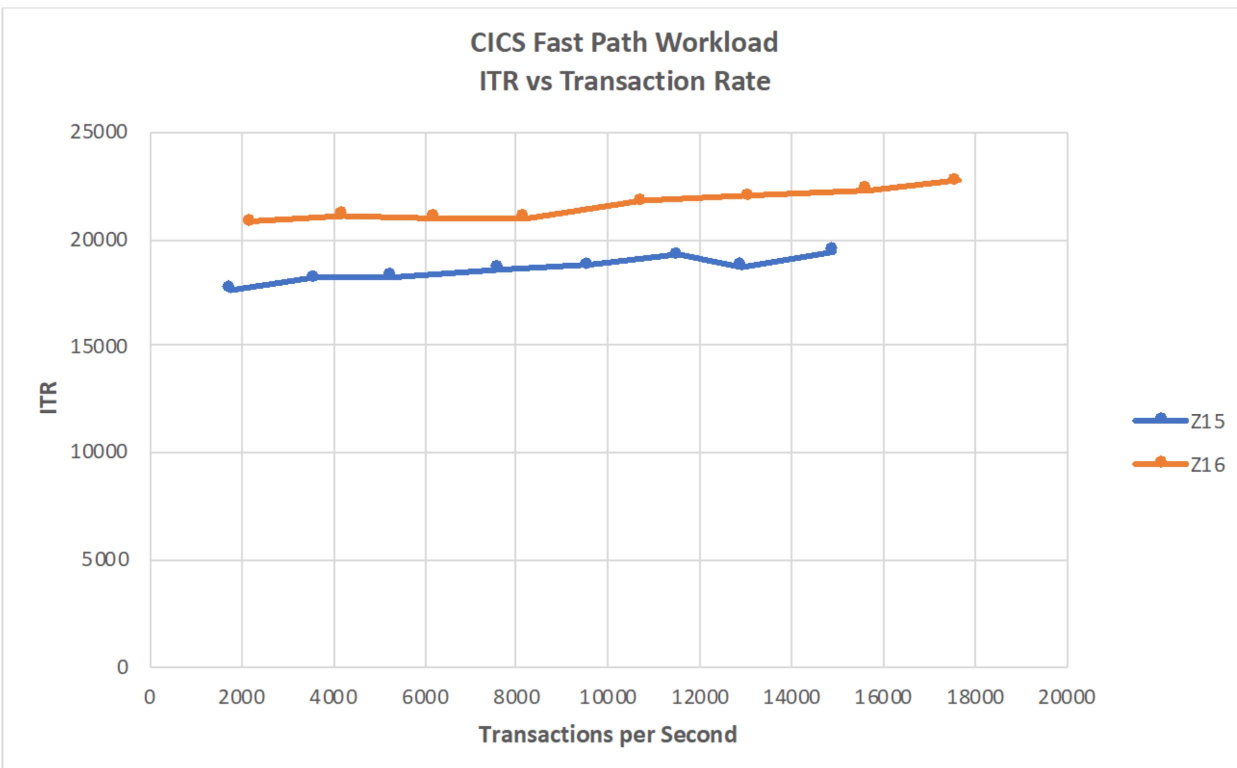


Figure 22: CICS IMS DBCTL Fast Path ITR versus Transaction Rate Comparison Results

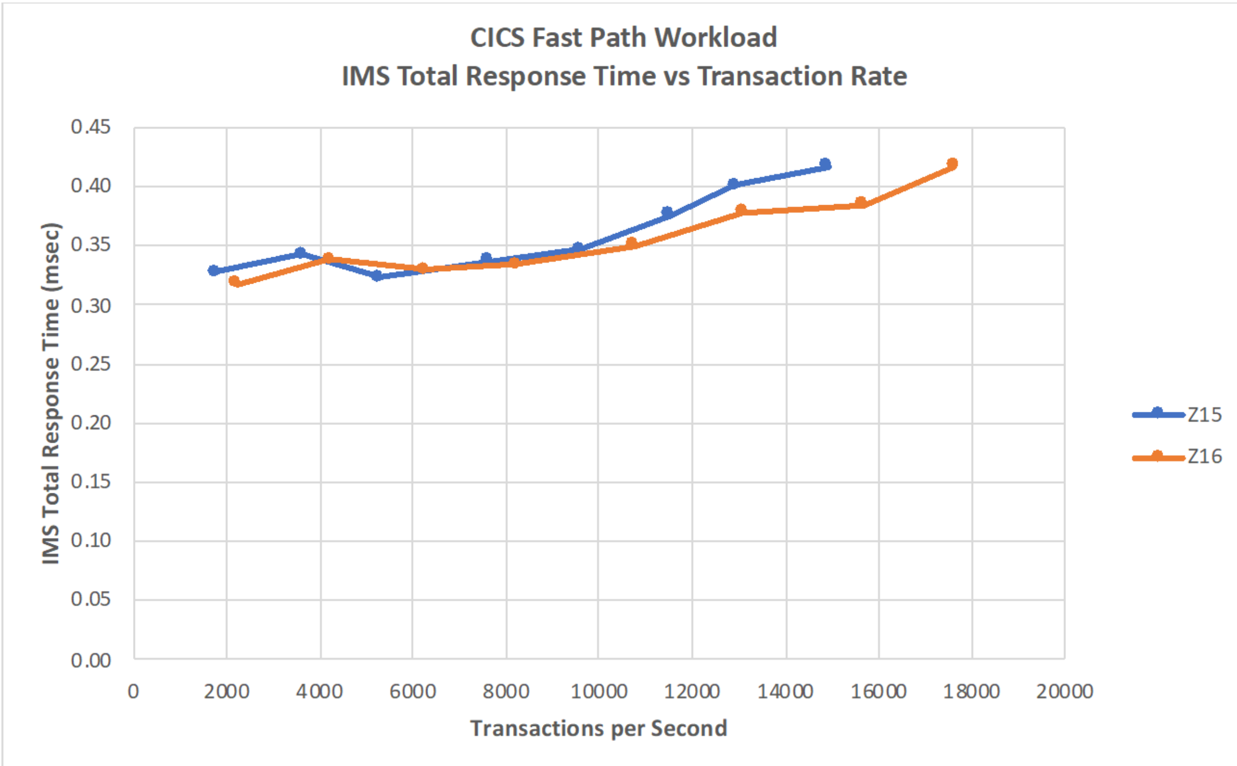


Figure 23: CICS IMS DBCTL Fast Path IMS Total Response Time versus Transaction Rate Comparison Results

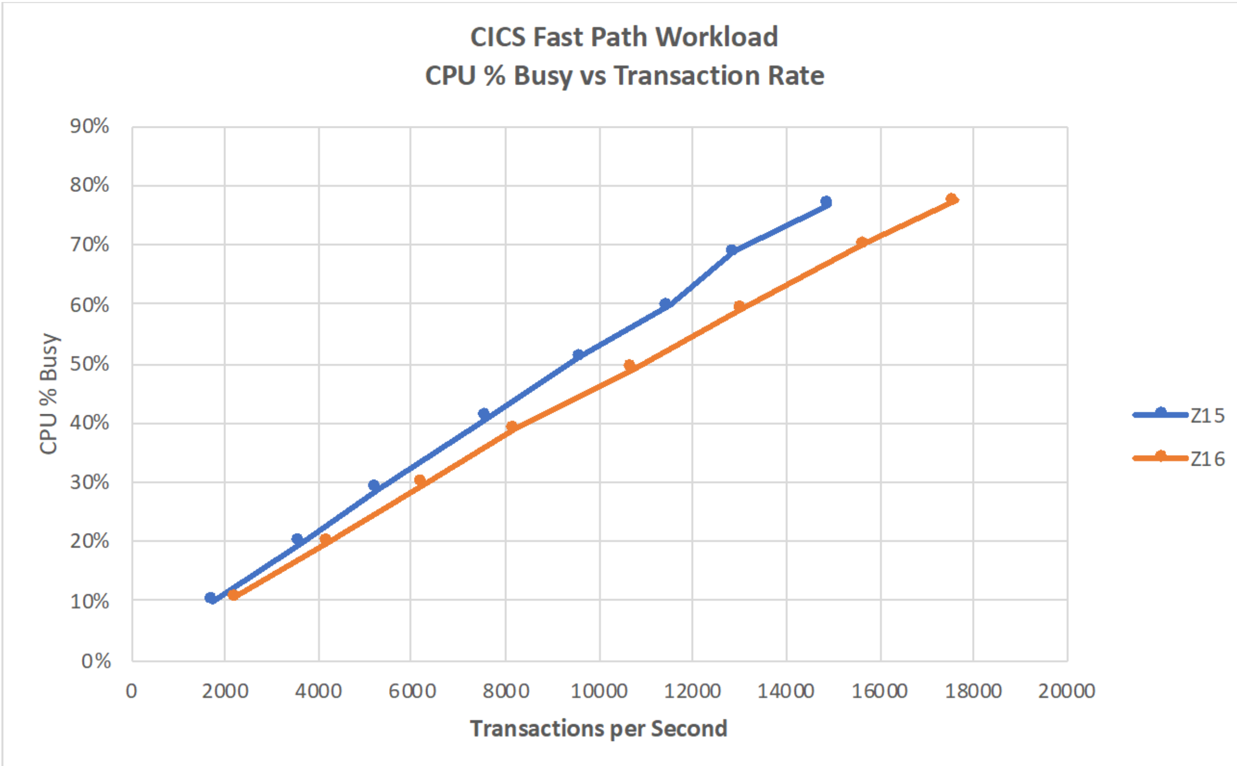


Figure 24: CICS IMS DBCTL Fast Path CPU % Busy versus Transaction Rate Comparison Results

6.6 IMS TM-Db2 IRWW Performance Evaluation

IMS provides access to external Db2 for z/OS databases through the External Subsystem Attach Facility (ESAF) by acting as the transaction manager (TM). The IMS TM-Db2 IBM Retail Warehouse Workload (IRWW) performance testing used ESAF to access a Db2 for z/OS relational database, which is based on a retail type of environment.

6.6.1 System Configuration

The IMS TM-Db2 IRWW base evaluation processes the IMS transactions in 240 MPP regions accessing the Db2 for z/OS retail warehouse database through ESAF. The workload was executed on a z14 configured in a two-LPAR environment as shown in Figure 25:

- LPAR 1 hosts Db2, IMS, 240 MPP regions, and ESAF using five general purpose engines
- LPAR 2 hosts a Java based workload driver driving 100 clients via TCPIP and four general purpose engines

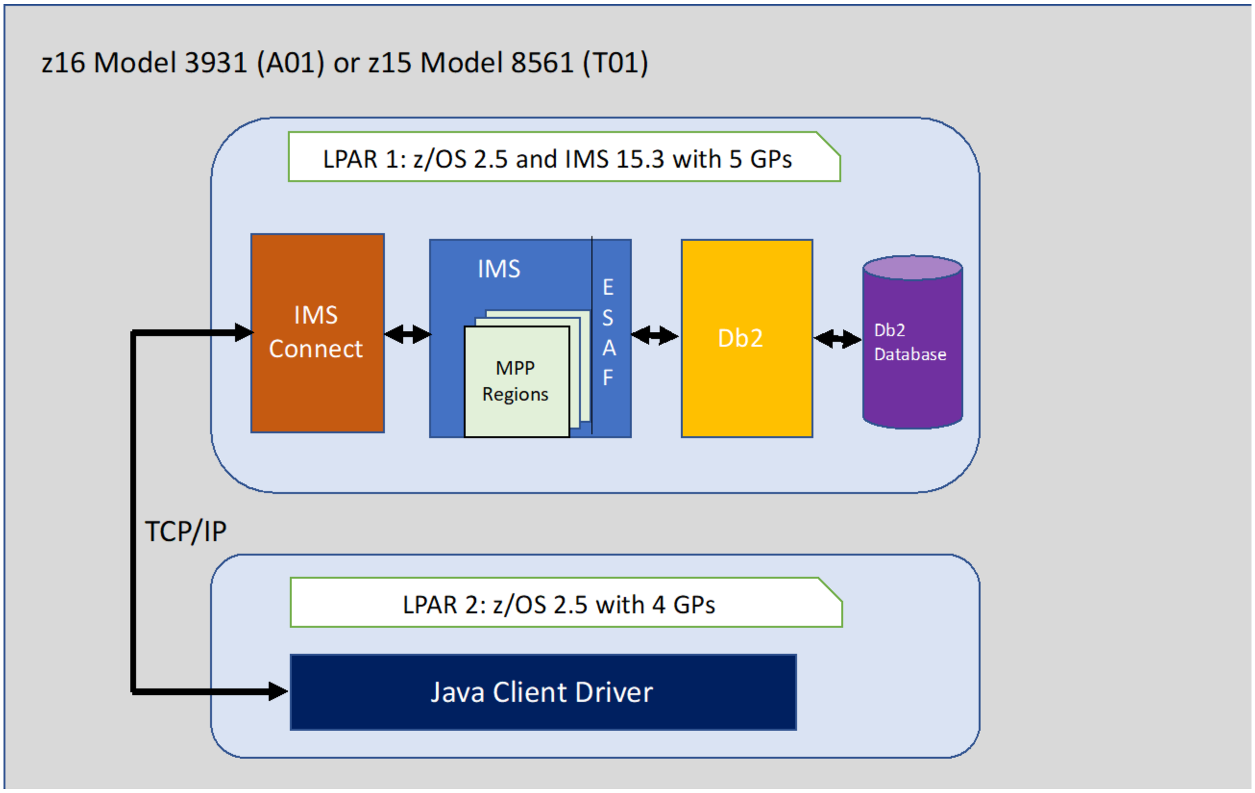


Figure 25: IMS TM-Db2 Configuration

6.6.2 Evaluation Results

The z16 showed an improvement in ETR and ITR at similar CPU usage over z15 for the IMS TM-Db2 IRWW workload. Table 25 shows the comparison between z16 and z15.

Table 25: IMS TM-Db2 Evaluation Results

IMS-Db2 Workload Evaluation				
	IBM z15	IBM z16	Delta	Delta %
CPU % Busy	80.03%	80.54%	0.51%	0.64%
ETR (Tran/Sec)	6782.32	7439.82	657.50	9.69%
ITR	8474.72	9237.42	762.70	9.00%

Total IMS Response Time (msec)	4.463	4.619	0.156	3.50%
Total General CPU µsec/Tran	589.990	541.277	-48.713	-8.26%
IMS CPU Service Time/Tran (µsec)	16.046	13.088	-2.958	-18.43%
ICON CPU Service Time/Tran (µsec)	15.958	14.361	-1.597	-10.01%
MPP CPU Service Time/Tran (µsec)	367.834	339.021	-28.813	-7.83%
Common Storage Below and Above 16 MB for Avg. Key 7				
Avg. CSA Below 16M Key 7 (K)	324	324	0	0.00%
Avg. CSA Above 16M Key 7 (M)	46.1	46.4	0.3	0.65%
Private Storage IMS Control Region				
Avg. LSQA Private (K)	676	676	0	0.00%
Avg. LSQA EPrivate (M)	14.9	15.0	0.1	0.67%
Avg. USER Private (K)	2384	2384	0	0.00%
Avg. USER EPrivate (M)	57.3	57.2	-0.1	-0.17%
Private Storage IMS DL/I Region				
Avg. LSQA Private (K)	348	348	0	0.00%
Avg. LSQA EPrivate (M)	9.3	9.3	0.036	0.39%
Avg. USER Private (K)	636	636	0	0.00%
Avg. USER EPrivate (M)	264	264	0	0.00%
Private Storage IMS Connect Region				
Avg. LSQA Private (K)	376	376	0	0.00%

Avg. LSQA EPrivate (M)	14.1	13.9	-0.2	-1.42%
Avg. USER Private (K)	56	56	0	0.00%
Avg. USER EPrivate (M)	177	177	0	0.00%

A series of scaling tests were run to compare z16 against z15 at various CPU percent busy values. Figure 26 shows the ITR versus transaction rate comparison, and Figure 27 shows the IMS response time versus transaction rate comparison. Figure 28 shows the CPU percent busy versus transaction rate demonstrating lower CPU cost for the z16 at various transaction rates.

The IMS TM-Db2 IRWW workload on z16 showed the following improvements over z15:

- Up to 9% ITR improvement for z16 as compared to z15
- Reduction of 18.4% in total IMS CPU service time per transaction in z16
- Improved Total General CPU service time per transaction of almost 8.2% in z16
- Lower CPU usage for z16 compared to z15

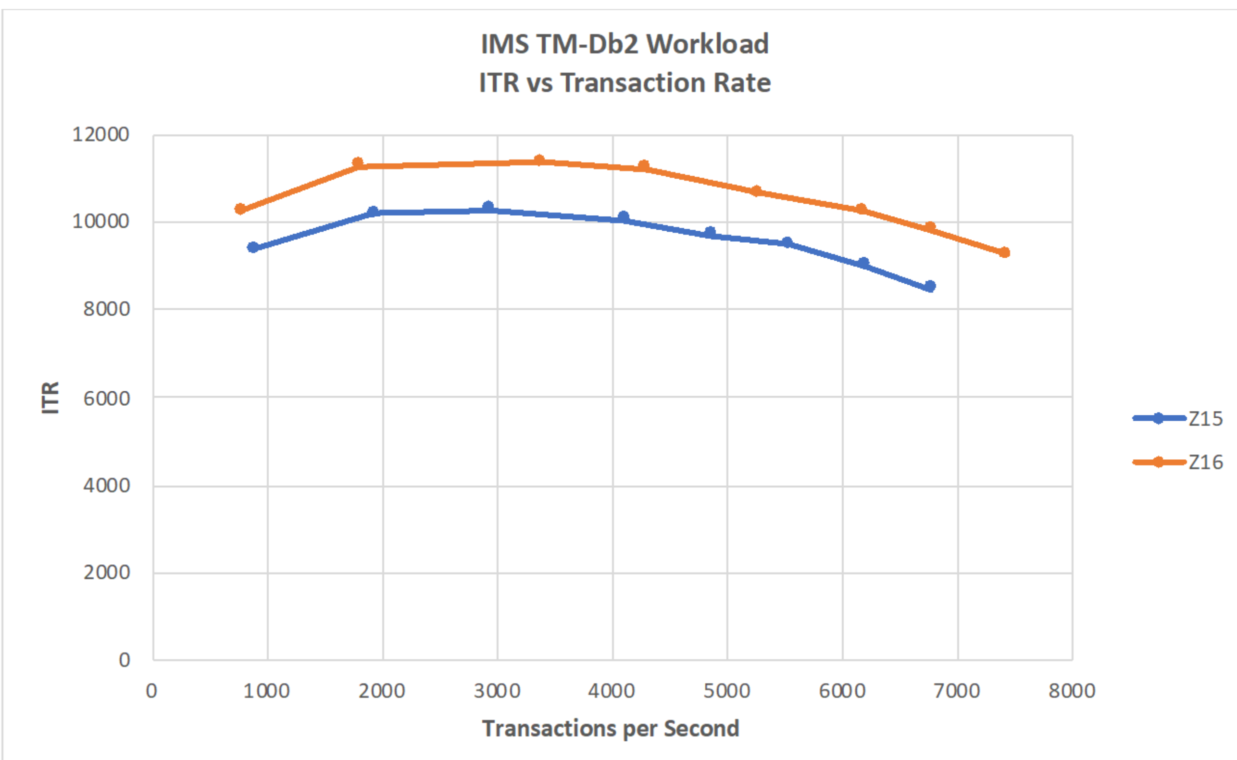


Figure 26: IMS TM-Db2 ITR versus Transaction Rate Comparison Results

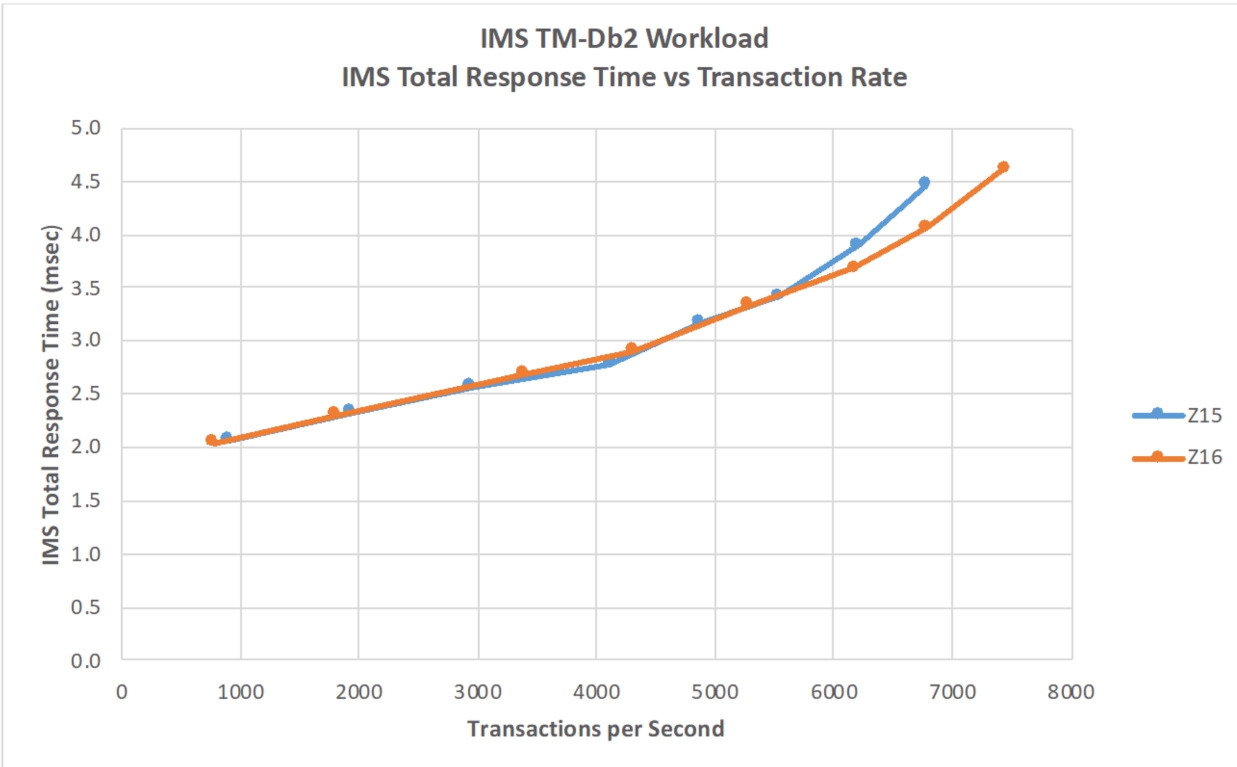


Figure 27: IMS TM-Db2 IMS Total Response Time versus Transaction Rate Comparison Results

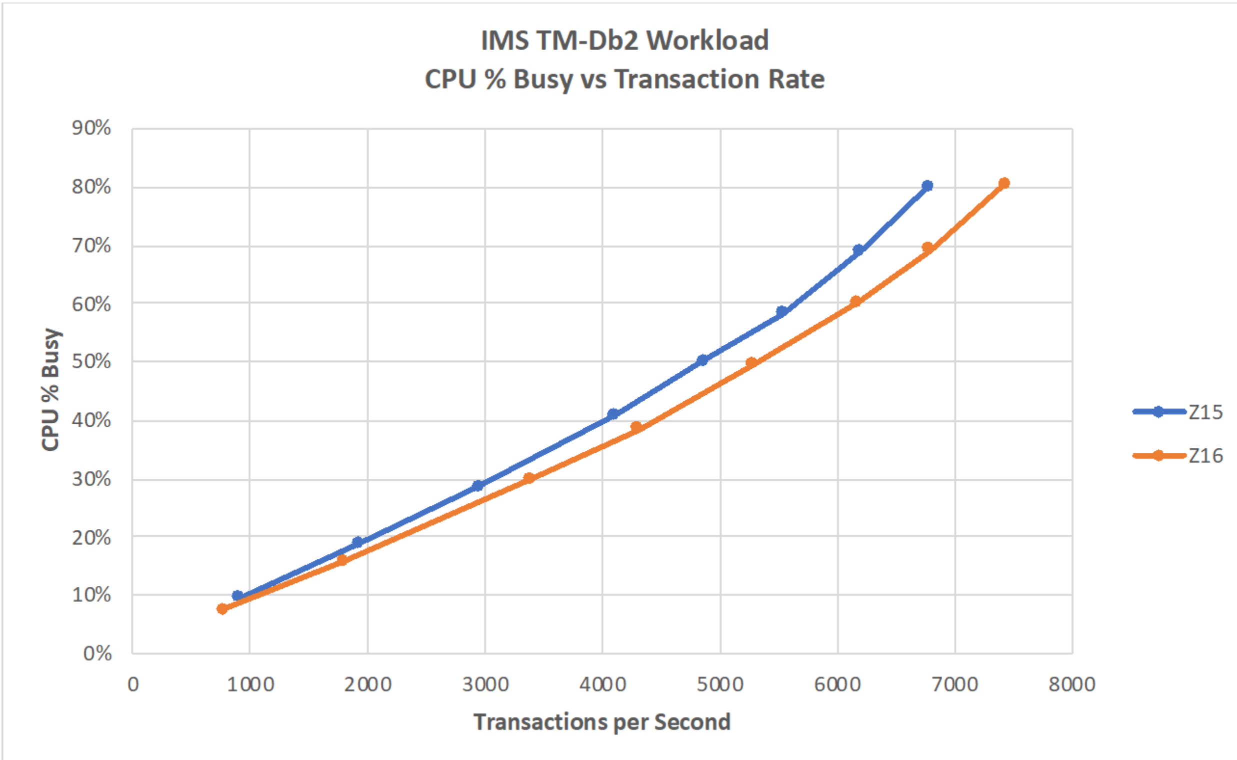


Figure 28: IMS TM-Db2 CPU % Busy versus Transaction Rate Comparison Results

6.7 z/OS Connect with IMS Service Provider (SP) Performance Evaluation

The objective of the z/OS Connect with IMS SP evaluation was to compare the ITR between z16 and z15 in the same software configuration.

6.7.1 System Configuration

The z/OS Connect with IMS SP evaluation was executed on both z16 and z15 in a two-LPAR configuration as shown in Figure 29:

- LPAR 1 hosts z/OS Connect with IMS SP, IMS, 350 MPP regions, and IMS Connect with eight general purpose engines
- LPAR 2 hosts a Java-based workload driver driving 80 clients with six general purpose engines

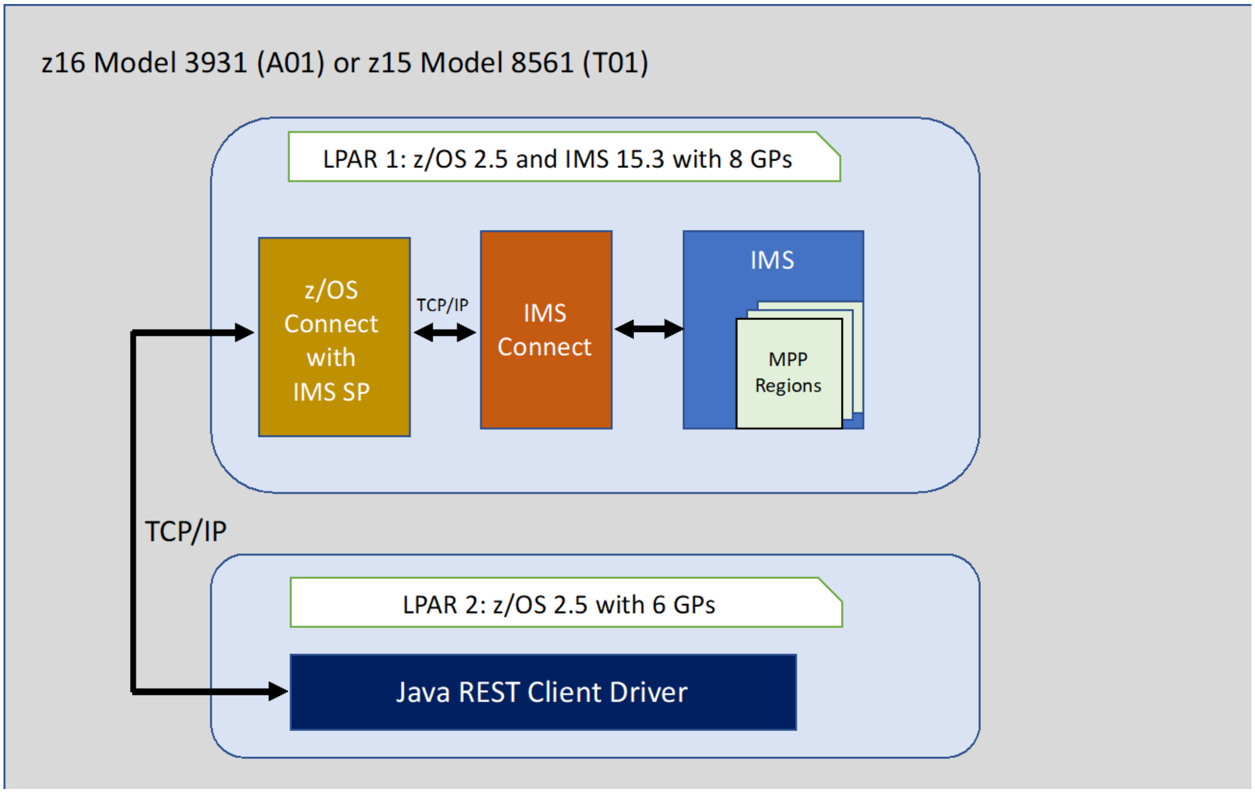


Figure 29: z/OS Connect with IMS SP Workload Environment Configuration

6.7.2 Evaluation Results

The z16 showed an improvement in ETR and ITR at similar CPU usage over z15 for the z/OS Connect with IMS SP workload. Table 26 shows the comparison between z16 and z15 for GPs.

Table 26: IMS Service Provider with General Purpose Engine Comparison Results

z/OS Connect with IMS Service Provider Workload Evaluation				
	IBM z15	IBM z16	Delta	Delta %
CPU % Busy	79.97%	77.37%	-2.60%	-3.25%
ETR (Tran/Sec)	23091.27	24915.96	1824.69	7.90%
ITR	28874.92	32203.64	3328.72	11.53%
Total IMS Response Time (msec)	0.163	0.130	-0.033	-20.25%

Total General CPU µsec/Tran	277.057	248.419	-28.638	-10.34%
IMS CPU Service Time/Tran (µsec)	11.158	9.161	-1.997	-17.90%
ICON CPU Service Time/Tran (µsec)	16.7	15.8	-0.9	-5.62%
MPP CPU Service Time/Tran (µsec)	21.217	18.504	-2.713	-12.79%
zCEE CPU Service Time/Tran (µsec)	211.698	191.546	-20.152	-9.52%
Common Storage Below and Above 16 MB for Avg. Key 7				
Avg. CSA Below 16M Key 7 (K)	332	332	0	0.00%
Avg. CSA Above 16M Key 7 (M)	24.8	25.8	1	4.03%
Private Storage IMS Control Region				
Avg. LSQA Private (K)	604	604	0	0.00%
Avg. LSQA EPrivate (M)	14.7	14.7	0	0.00%
Avg. USER Private (K)	2276	2272	-4	-0.18%
Avg. USER EPrivate (M)	56.5	56.8	0.3	0.53%
Private Storage IMS DL/I Region				
Avg. LSQA Private (K)	348	348	0	0.00%
Avg. LSQA EPrivate (M)	9.30	9.28	-0.02	-0.22%
Avg. USER Private (K)	636	636	0	0.00%
Avg. USER EPrivate (M)	264	264	0	0.00%
Private Storage ICON Region				
Avg. LSQA Private (K)	376	376	0	0.00%
Avg. LSQA EPrivate (M)	14.2	14.2	0	0.00%
Avg. USER Private (K)	56	56	0	0.00%

Avg. USER EPrivate (M)	176	176	0	0.00%

A series of scaling tests were run to compare z16 against z15 at various CPU percent busy values with GPs only. Figure 30 shows the ITR versus transaction rate comparison, and Figure 31 shows the IMS response time versus transaction rate comparison. Figure 32 shows the CPU percent busy versus transaction rate demonstrating lower CPU usage for z16 at various transaction rates.

The z/OS Connect with IMS SP workload on z16 showed the following improvements over z15 with GPs:

- Up to 11.5% ITR improvement for z16 as compared to z15
- Improved IMS transaction response time of 20.25% in z16
- Reduction of about 18% in total IMS CPU service time per transaction in z16
- Lower CPU usage for z16 compared to z15

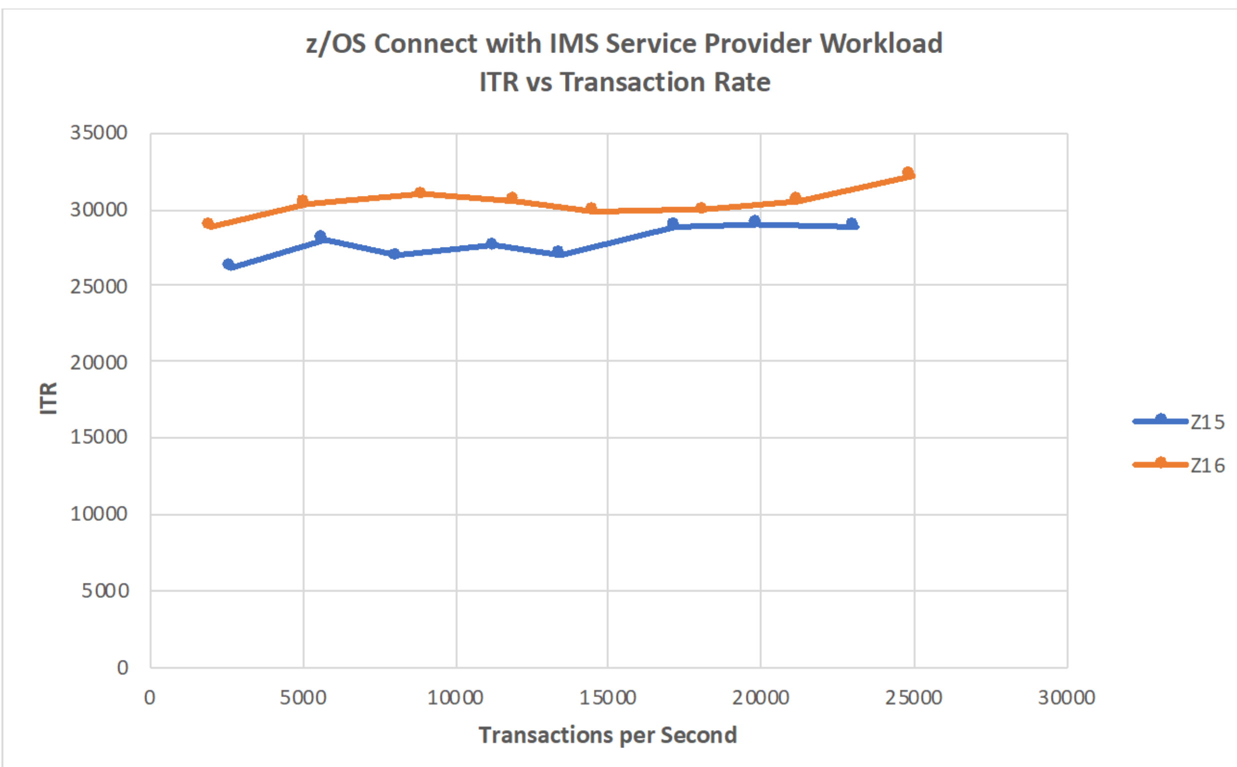


Figure 30: z/OS Connect with IMS Service Provider ITR versus Transaction Rate Comparison

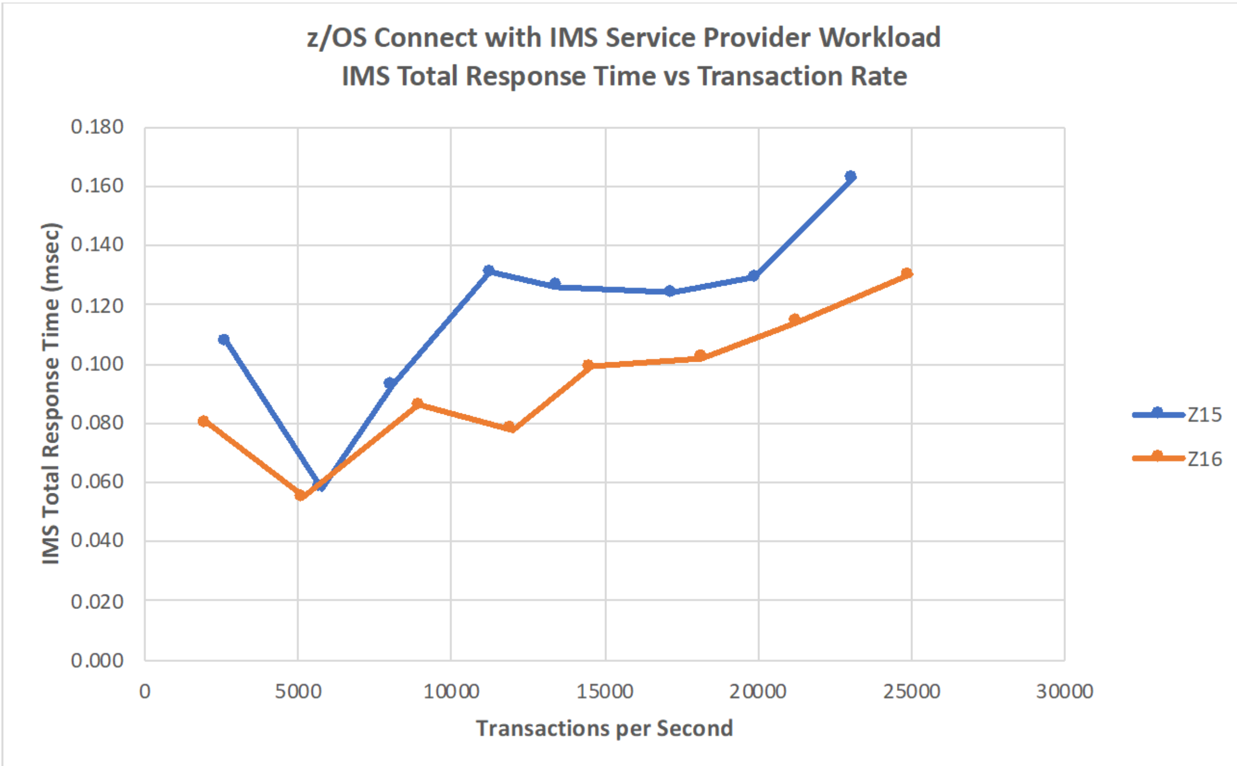


Figure 31: z/OS Connect with IMS Service Provider IMS Total Response Time versus Transaction Rate Comparison

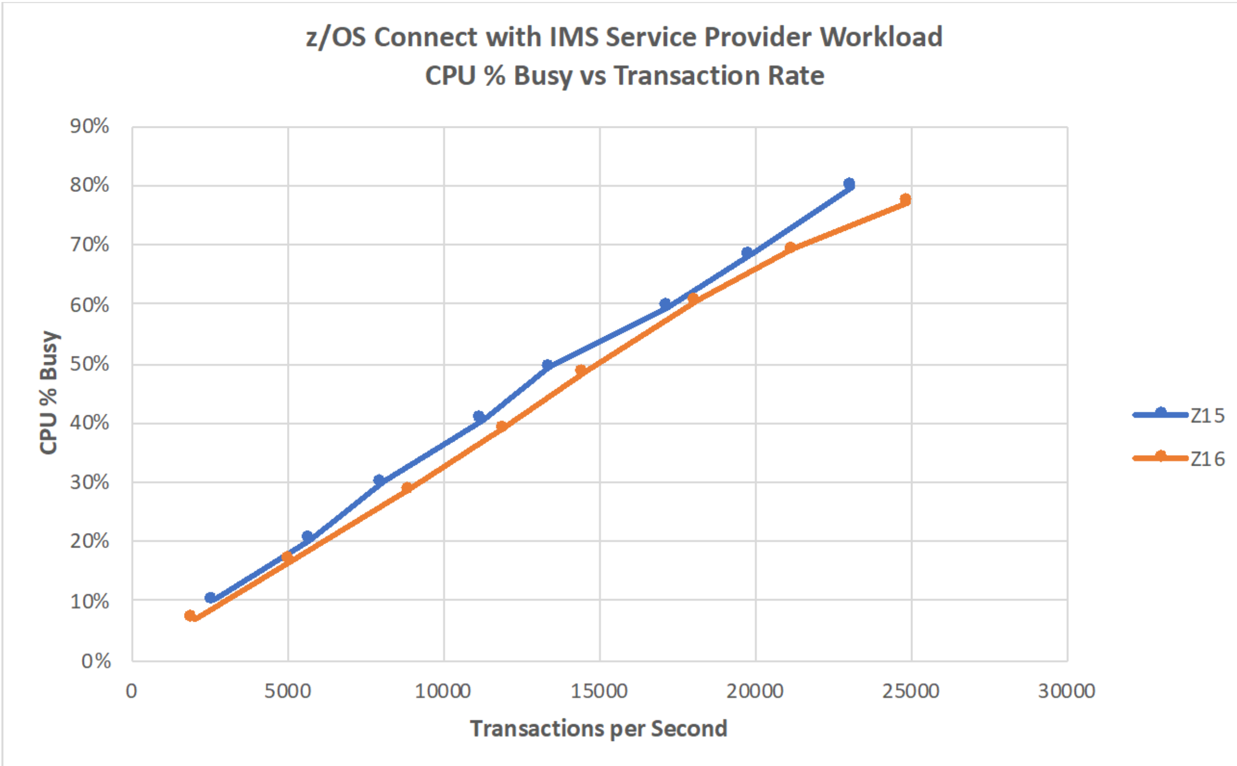


Figure 32: z/OS Connect with IMS Service Provider CPU % Busy versus Transaction Rate Comparison

6.8 Java Message Processing (JMP) Performance Evaluation

The objective of the JMP evaluation was to compare the ITR between z16 and z15 in the same software configuration.

6.8.1 System Configuration

The JMP evaluation was executed on both z16 and z15 in a two-LPAR configuration as shown in Figure 33:

- LPAR 1 hosts IMS, IMS Connect and 50 JMP regions with five general purpose engines
- LPAR 2 hosts a Java workload driver driving 60 clients with six general purpose engines

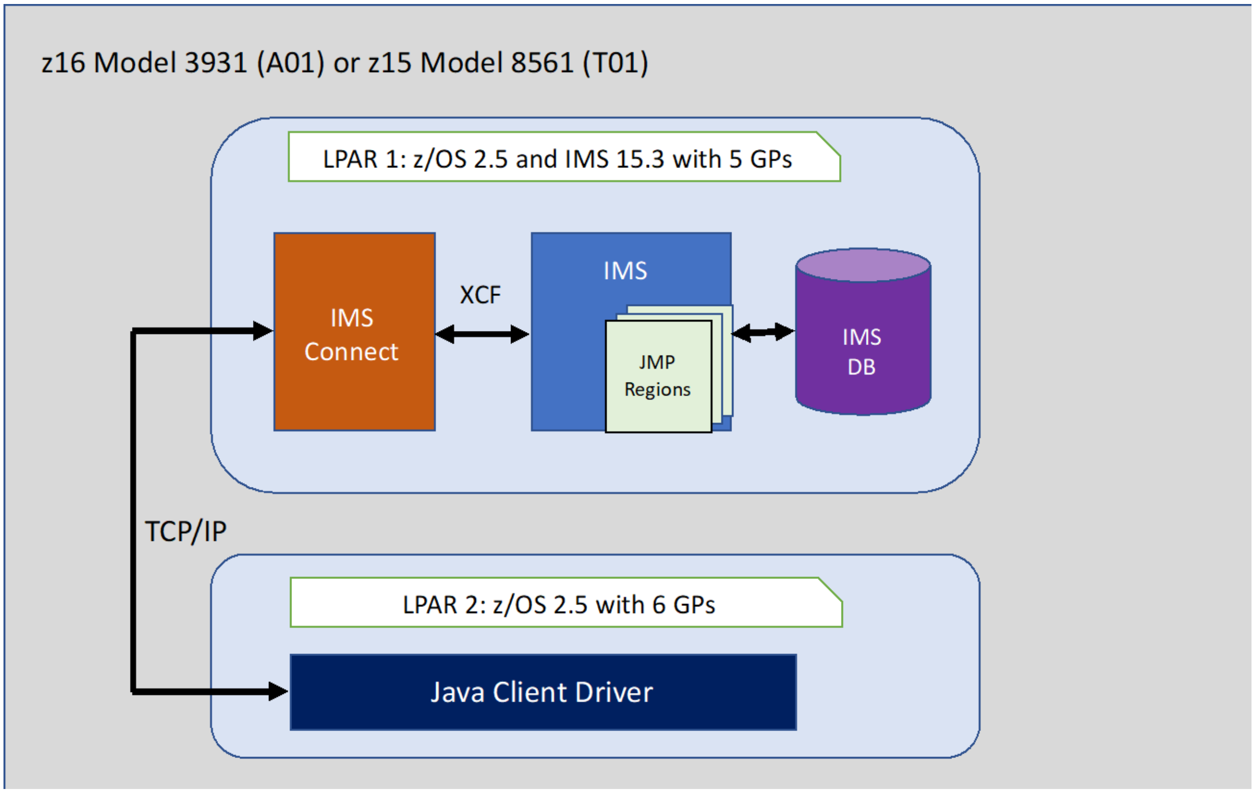


Figure 33: JMP Environment Configuration

6.8.2 Evaluation Results

The z16 showed an improvement in ETR and ITR at similar CPU usage over z15 for the JMP workload. Table 27 shows the comparison between z16 and z15 for GPs.

Table 27: IMS JMP with General Purpose Engine Comparison Results

Java Message Processing Workload Evaluation				
	IBM z15	IBM z16	Delta	Delta %
CPU % Busy	81.29%	79.47%	-1.82%	-2.24%
ETR (Tran/Sec)	35899.70	38608.32	2708.62	7.54%
ITR	44162.50	48582.26	4419.76	10.01%
Total IMS Response Time (msec)	0.628	0.538	-0.090	-14.33%

Total General CPU µsec/Tran	113.218	102.918	-10.300	-9.10%
IMS CPU Service Time/Tran (µsec)	11.004	10.606	-0.398	-3.62%
ICON CPU Service Time/Tran (µsec)	14.092	12.998	-1.094	-7.76%
JMP CPU Service Time/Tran (µsec)	73.022	66.155	-6.867	-9.40%
Common Storage Below and Above 16 MB for Avg. Key 7				
Avg. CSA Below 16M Key 7 (K)	236	236	0	0.00%
Avg. CSA Above 16M Key 7 (M)	17.6	17.7	0.1	0.57%
Private Storage IMS Control Region				
Avg. LSQA Private (K)	604	604	0	0.00%
Avg. LSQA EPrivate (M)	14.6	14.6	0	0.00%
Avg. USER Private (K)	2276	2270	-6	-0.26%
Avg. USER EPrivate (M)	56.9	57.1	0.2	0.35%
Private Storage IMS DL/I Region				
Avg. LSQA Private (K)	380	380	0	0.00%
Avg. LSQA EPrivate (M)	9.42	9.43	0.01	0.11%
Avg. USER Private (K)	640	640	0	0.00%
Avg. USER EPrivate (M)	264	264	0	0.00%
Private Storage IMS Connect Region				
Avg. LSQA Private (K)	376	376	0	0.00%
Avg. LSQA EPrivate (M)	14.2	14.2	0	0.00%

Avg. USER Private (K)	56	56	0	0.00%
Avg. USER EPrivate (M)	177	177	0	0.00%

A series of scaling tests were run to compare z16 against z15 at various CPU percent busy values with GPs. Figure 34 shows the ITR versus transaction rate comparison, and Figure 35 shows the IMS response time versus transaction rate comparison. Figure 36 shows the CPU percent busy versus transaction rate demonstrating lower CPU usage for the z16 at various transaction rates.

The JMP workload on z16 showed the following improvements over z15 with GPs:

- Up to 10% ITR improvement for z16 as compared to z15
- Improved IMS transaction response time of 14.3% in z16
- Reduction in total IMS CPU service time per transaction in z16
- Reduction in JMP CPU service time per transaction of 9.4% in z16
- Lower CPU usage for z16 compared to z15

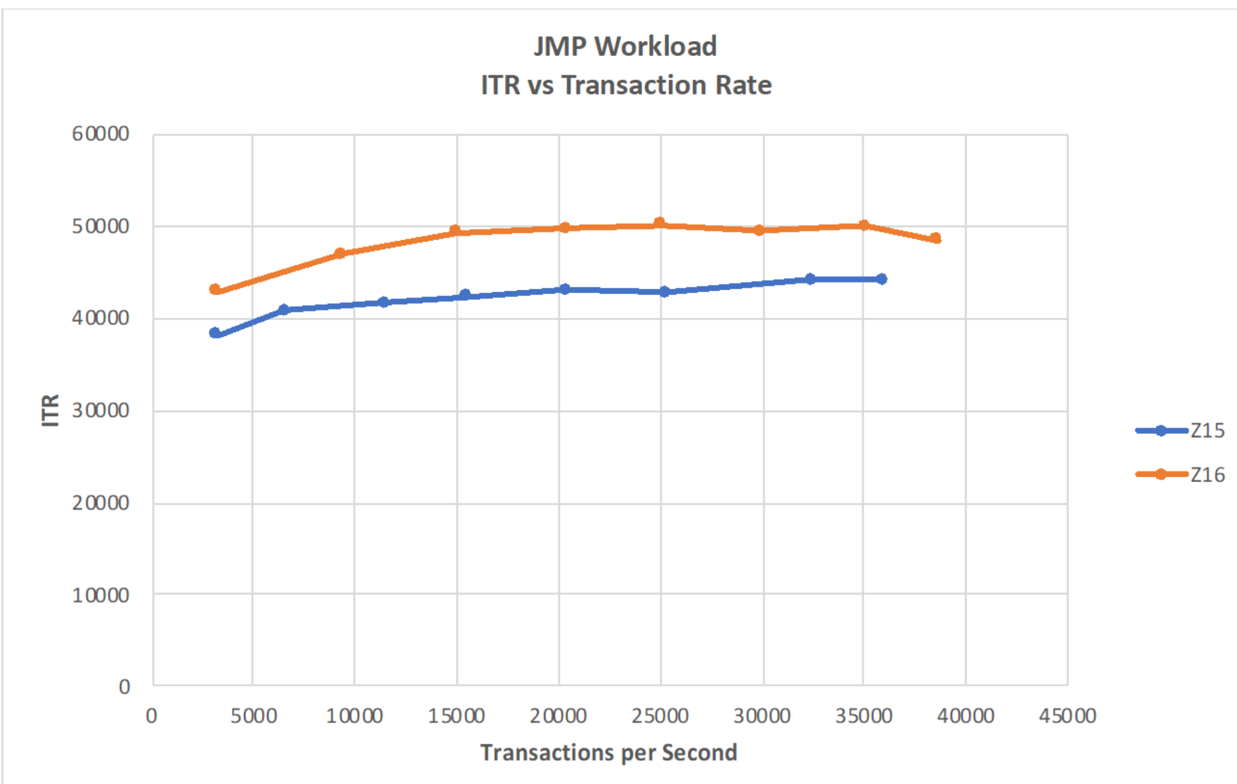


Figure 34: JMP ITR versus Transaction Rate Comparison Results

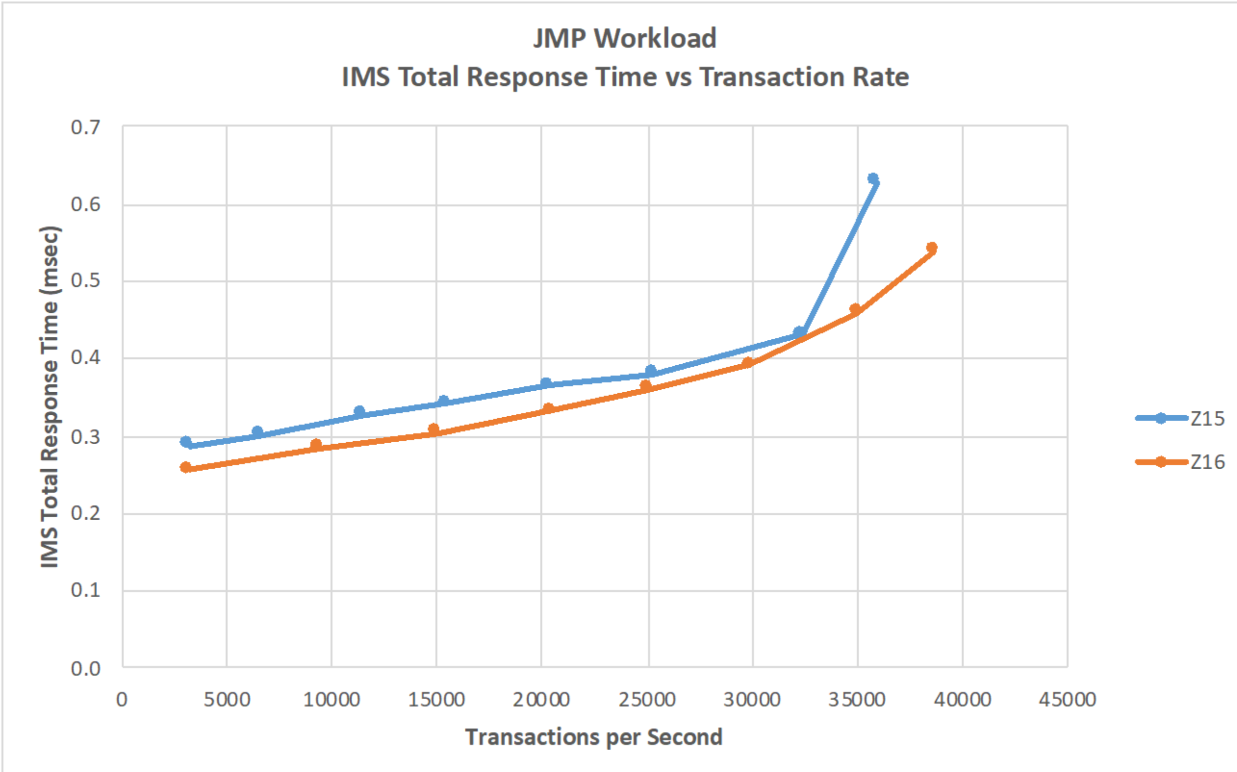


Figure 35: JMP IMS Total Response Time versus Transaction Rate Comparison Results

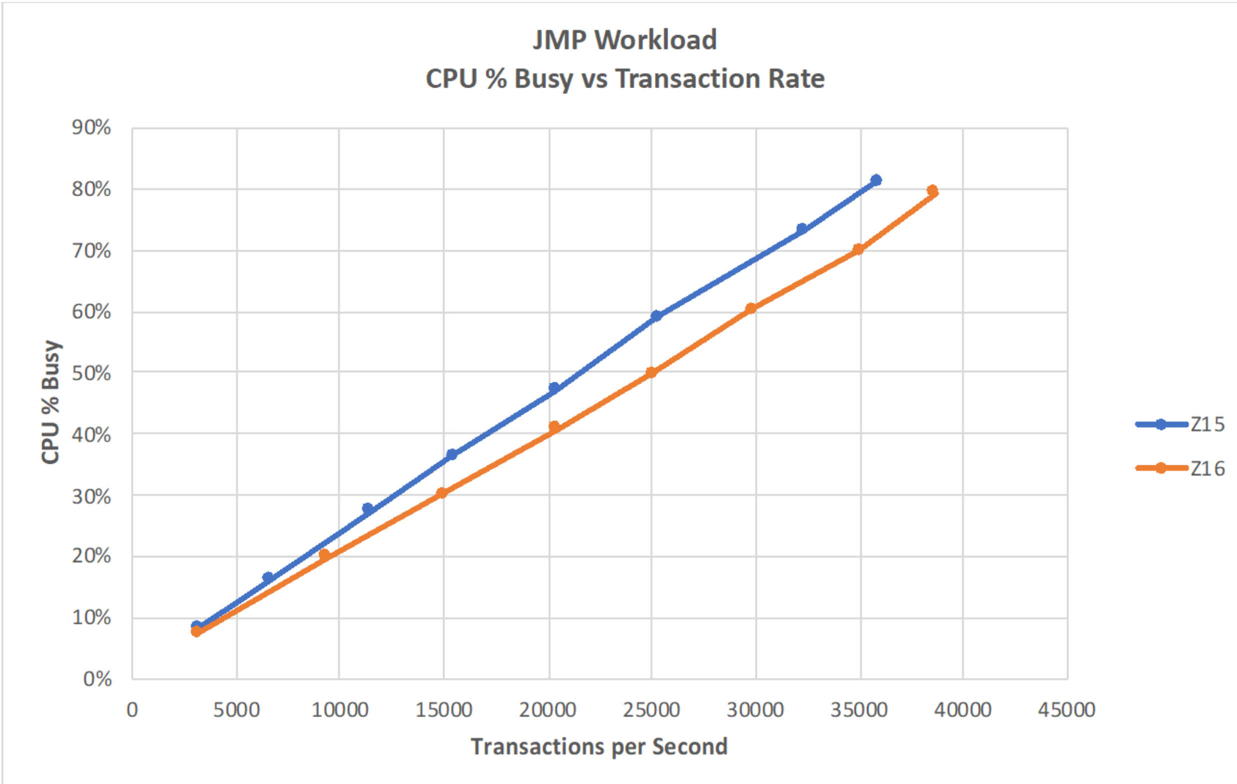


Figure 36: JMP CPU % Busy versus Transaction Rate Comparison Results

6.9 Open Database Management (ODBM) Performance Evaluation

ODBM is a CSL region that manages database connections and access requests from application programs that use the following resource adapters and APIs:

- IMS Universal Database resource adapter
- MS Universal JDBC driver
- IMS Universal DL/I driver
- Open Database Access interface (ODBA)
- ODBM CSLDMI interface

For the ODBM workload performance measurement, the IMS Universal JDBC driver was used to communicate with ODBM through IMS Connect, using the open standard Distributed Relational Database Architecture (DRDA) as the low-level communication protocol using the Distributed Data Management (DDM) architecture. The ODBM translates the DDM into DL/I calls and packages the IMS output as DDM to be returned to the client. Also, the ODBM can run with or without z/OS Resource Recovery Services (RRS), however, by default, ODBM runs with RRS.

6.9.1 System Configuration

The JMP evaluation was executed on both z16 and z15 in a two-LPAR configuration as shown in Figure 37.

- LPAR 1 hosts IMS, ODBM, and IMS Connect with four general purpose engines
- LPAR 2 hosts a Java-based workload driver driving 40 clients via TCP/IP with four general purpose engines

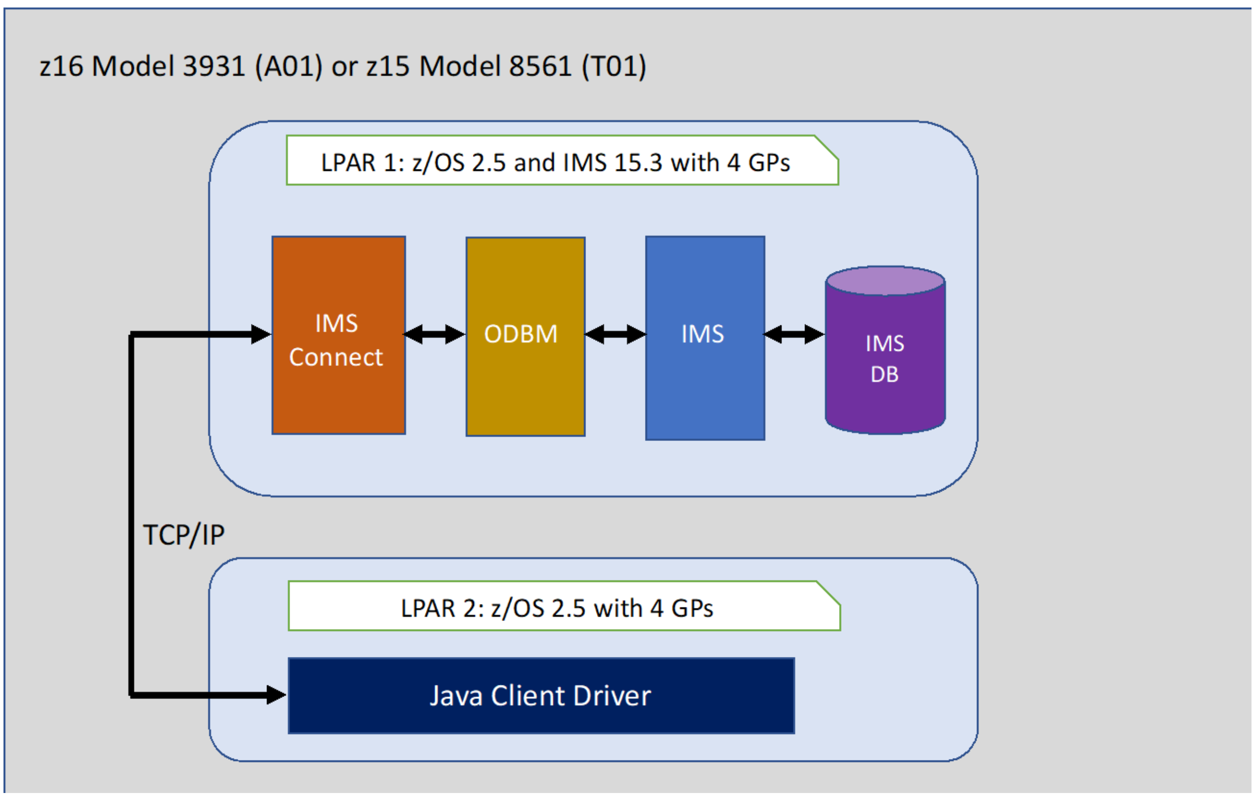


Figure 37: ODBM Processing Configuration

6.9.2 Evaluation Results

The z16 showed an improvement in ETR and ITR at similar CPU usage over z15 for the JMP workload Table 28 shows the comparison between z16 and z15.

Table 28: ODBM Comparison Results

ODBM Workload Evaluation				
	IBM z15	IBM z16	Delta	Delta %
CPU % Busy	77.37%	75.68%	-1.69%	-2.18%
ETR (Tran/Sec)	6586.98	7395.66	808.68	12.28%
ITR	8513.61	9772.28	1258.67	14.78%
Total IMS Response Time (msec)	4.311	4.413	0.102	2.37%
Total General CPU µsec/Tran	469.836	409.321	-60.515	-12.88%
IMS CPU Service Time/Tran (µsec)	13.932	12.921	-1.011	-7.26%
ICON CPU Service Time/Tran (µsec)	141.942	129.963	-11.979	-8.44%
ODBM CPU Service Time/Tran (µsec)	219.491	188.37	-31.121	-14.18%
Common Storage Below and Above 16 MB for Avg. Key 7				
Avg. CSA Below 16M Key 7 (K)	252	244	-8	-3.17%
Avg. CSA Above 16M Key 7 (M)	18.5	18.6	0.1	0.54%
Private Storage IMS Control Region				
Avg. LSQA Private (K)	608	608	0	0.00%
Avg. LSQA EPrivate (M)	14.3	14.3	0	0.00%
Avg. USER Private (K)	2276	2276	0	0.00%
Avg. USER EPrivate (M)	55.5	55.5	0	0.00%
Private Storage IMS DL/I Region				
Avg. LSQA Private (K)	380	380	0	0.00%

Avg. LSQA EPrivate (M)	9.48	9.45	-0.03	-0.32%
Avg. USER Private (K)	640	640	0	0.00%
Avg. USER EPrivate (M)	264	264	0	0.00%
Private Storage IMS Connect Region				
Avg. LSQA Private (K)	376	376	0	0.00%
Avg. LSQA EPrivate (M)	13.9	13.9	0	0.00%
Avg. USER Private (K)	56	56	0	0.00%
Avg. USER EPrivate (M)	180	180	0	0.00%

A series of scaling tests were run to compare z16 against z15 at various CPU percent busy values with GPs. Figure 38 shows the ITR versus transaction rate comparison, and Figure 39 shows the IMS response time versus transaction rate comparison. Figure 40 shows the CPU percent busy versus transaction rate demonstrating lower CPU usage for the z16 at various transaction rates.

The OBM workload on z16 showed the following improvements over z15 with GPs:

- Up to 14.7% ITR improvement for z16 as compared to z15
- Improved IMS transaction response time of 2.37% in z16
- Reduction of 7.26% in total IMS CPU service time per transaction in z16
- Reduction in ODBM CPU service time per transaction of 14.18% in z16
- Lower CPU usage for z16 compared to z15

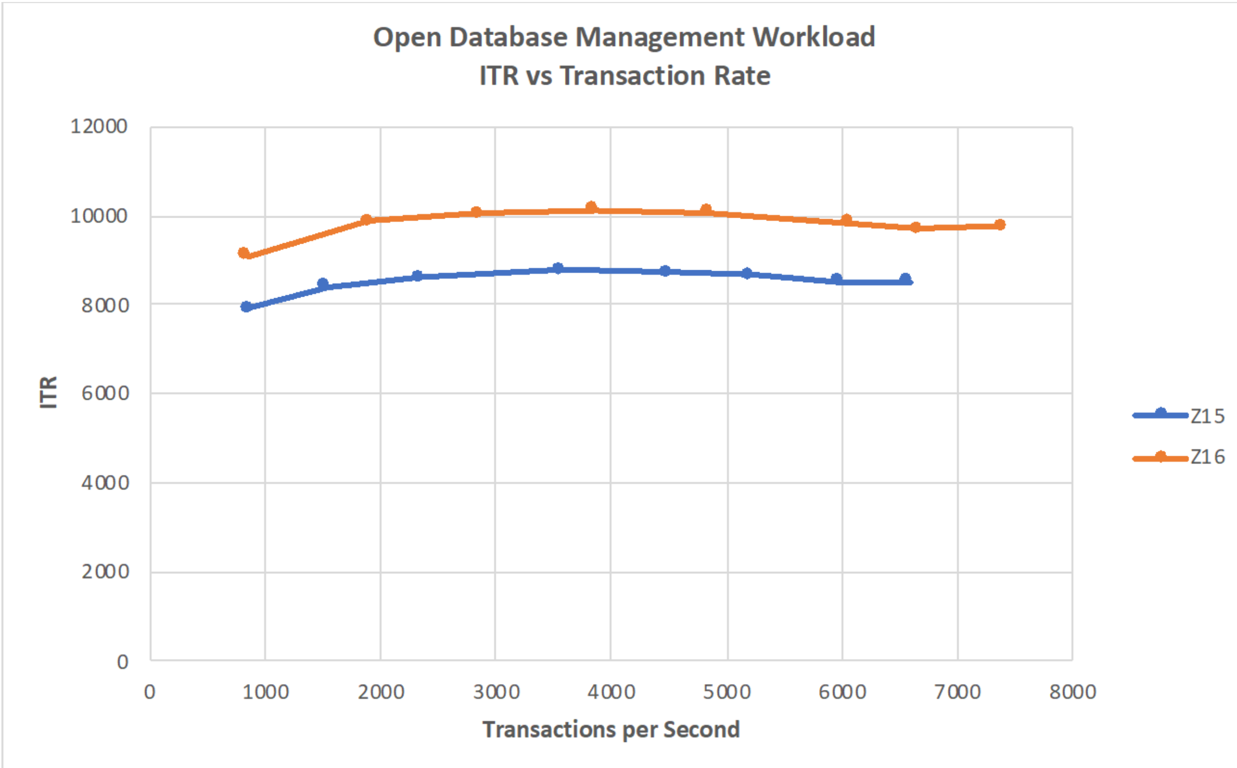


Figure 38: ODBM ITR versus Transaction Rate Comparison Chart

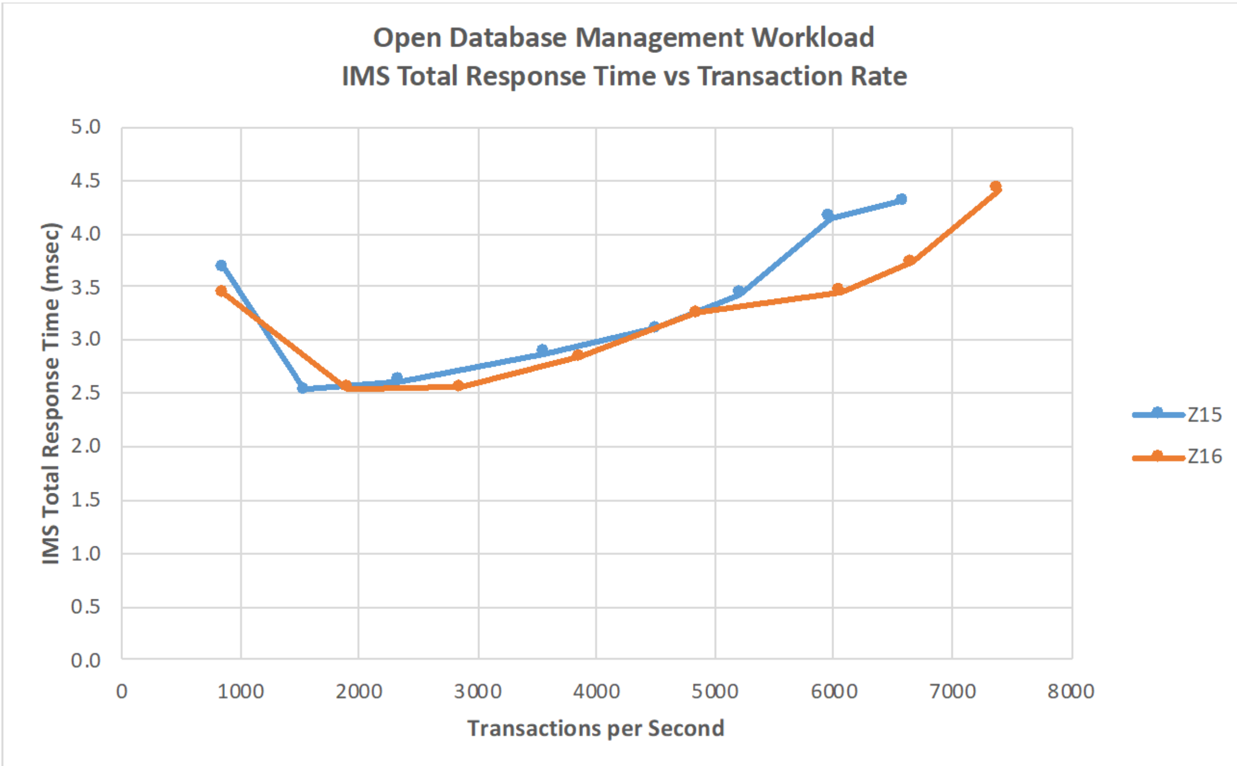


Figure 39: ODBM IMS Total Response Time versus Transaction Rate Comparison Chart

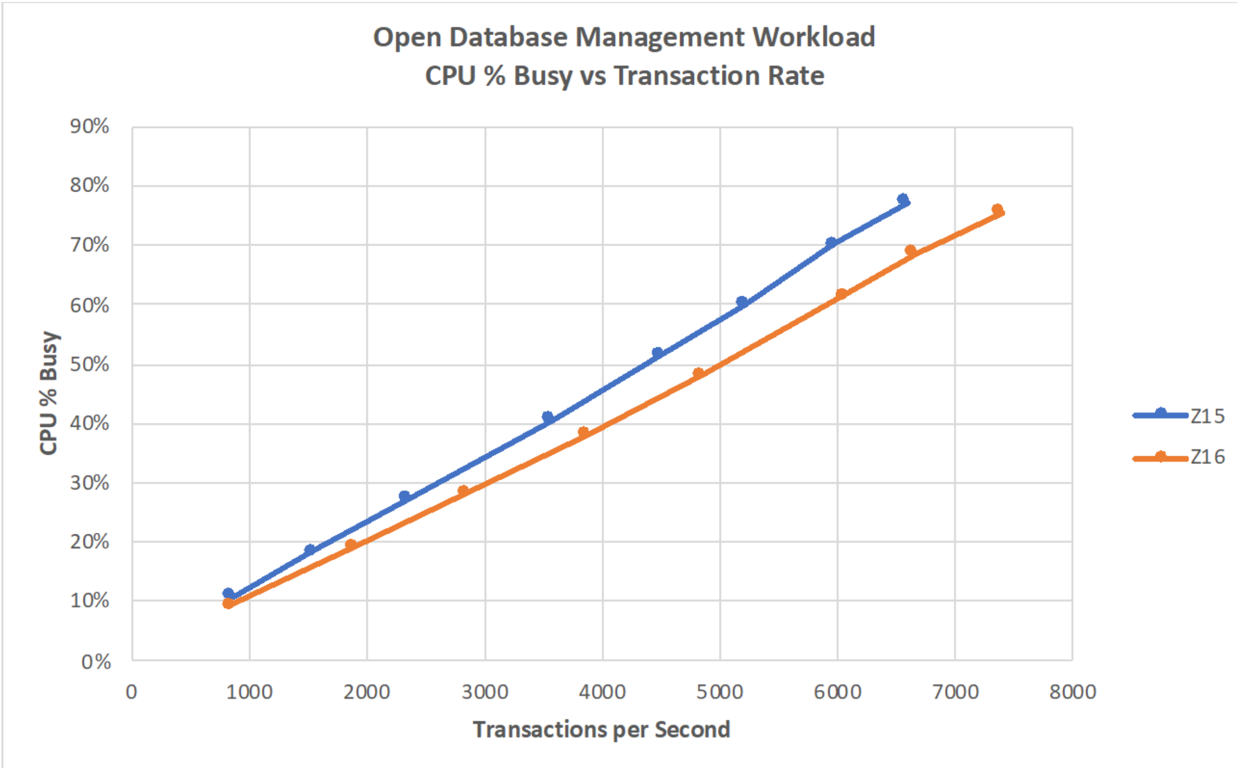


Figure 40: ODBM CPU % Busy versus Transaction Rate Comparison Chart

6.10 z16 Performance Summary

The results of the base performance workloads in our in-house environment yielded significant improvements in ITR ranging from 4% to 21% depending on the workload. Figure 41 shows a summary of the ITR improvements in z16 from z15 for each of the base performance workloads.

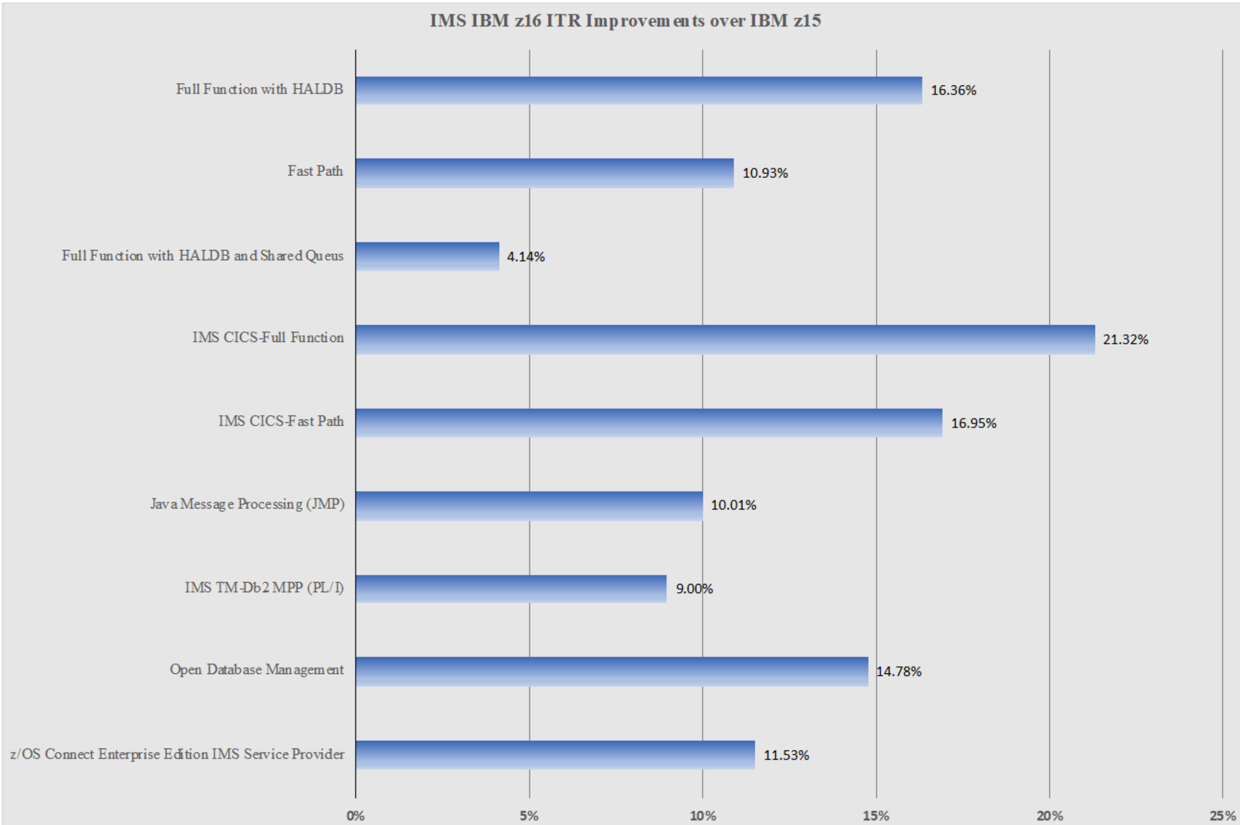


Figure 41: Performance Improvements in IBM z16 over IBM z15

7 Conclusion

The IBM z16 delivers excellent performance at similar CPU usage in all our IMS workloads as compared to the IBM z15. All our workloads showed an increase in both ITR and ETR with decreases in response and service time per transaction. In addition, the allocated storage for each of the workloads stayed relatively flat without any significant increase.

Overall, the IBM z16 continues to provide excellent performance as the newest generation of mainframe running the latest IMS version. IMS customers can continue to benefit and expect reliable, scalable, and increased performance with the z16.

8 Resources

IBM z16 Announce and Information:

- Announcement Letter: https://www.ibm.com/common/ssi/ShowDoc.wss?docURL=/common/ssi/rep_ca/1/760/ENUSJG22-0001/index.html
- Introduction Page: https://www.ibm.com/products/z16?mhsrc=ibmsearch_a&mhq=z16
- Press Release: https://newsroom.ibm.com/2022-04-05-Announcing-IBM-z16-Real-time-AI-for-Transaction-Processing-at-Scale-and-Industrys-First-Quantum-Safe-System
- Technical Specs: https://www.ibm.com/demos/it-infrastructure/vcms/en/5720708789305344/1649418775378.pdf?title=IBM%20z16%20model%20a01%20specification%20sheet
- IBM Redbooks®: https://www.redbooks.ibm.com/Redbooks.nsf/pages/z16

z/OS V2.5 Announcement Letter:
https://www.ibm.com/common/ssi/ShowDoc.wss?docURL=/common/ssi/rep_ca/9/872/ENUSAP21-0249/index.html

IMS 15.3 Information Center Link:
https://www.ibm.com/docs/en/ims/15.3.0

IMS V15.3 Announcement Letter:
https://www.ibm.com/common/ssi/ShowDoc.wss?docURL=/common/ssi/rep_ca/2/897/ENUS222-002/index.html&request_locale=en

IMS Home Page:
https://www.ibm.com/products/ims

Recommended Publication:
An Introduction to IMS - Your Complete Guide to IBM Information Management Systems

9 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 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.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Director of Licensing
IBM Corporation
North Castle Drive, MD-NC119
Armonk, NY 10504-1785
US

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

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 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.

Terms and conditions for product documentation

Permissions for the use of these publications are granted subject to the following terms and conditions.

Applicability

The terms and conditions are in addition to any terms of use for the IBM website.

Personal use

You may reproduce these publications for your personal, noncommercial use provided that all proprietary notices are preserved. You may not distribute, display or make derivative work of these publications, or any portion thereof, without the express consent of IBM.

Commercial use

You may reproduce, distribute and display these publications solely within your enterprise provided that all proprietary notices are preserved. You may not make derivative works of these publications, or reproduce, distribute or display these publications or any portion thereof outside of your enterprise, without the express consent of IBM.

Rights

Except as expressly granted in this permission, no other permissions, licenses or rights are granted, either express or implied, to the publications or any information, data, software or other intellectual property contained therein.

IBM reserves the right to withdraw the permissions granted herein whenever, in its discretion, the use of the publications is detrimental to its interest or, as determined by IBM, the above instructions are not being properly followed.

You may not download, export or re-export this information except in full compliance with all applicable laws and regulations, including all United States export laws and regulations.

IBM MAKES NO GUARANTEE ABOUT THE CONTENT OF THESE PUBLICATIONS. THE PUBLICATIONS ARE PROVIDED "AS-IS" AND WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO IMPLIED WARRANTIES OF MERCHANTABILITY, NON-INFRINGEMENT, AND FITNESS FOR A PARTICULAR PURPOSE.