IBM Technical Brief

IBM System zEnterprise®, System Storage®, and DB2® 10 for z/OS®: SAP® Banking Services 7.0 150 Million Accounts Measurements

Authors:

Seewah Chan
John W. Langer
Paul Lekkas
Rose L. Manz
Howard E. Poole
Michael R. Sheets

Document Owner:

Michael R. Sheets
IBM Americas Solutions Advanced Technical Skills

Version: 1.10
Date: February 22, 2012

Filename: System_z_SAP_SBS_150M.pdf

TABLE OF CONTENTS

TABLES

FIGURES

1 Introduction

Bank posting and balancing operations (discussed in more detail in Section 6, "Workload and Measurement Background" on page 7) are often called core banking. These processes are crucial to all commercial banks. Historically, core banking was one of the first parts of a banking operation automated with computers. Not only is it crucial to the bank's business, but it generally has the highest processing volumes and resource requirements and has extremely high availability requirements. According to some sources, core banking was done mainly on mainframes – mostly as custom developed CICS or IMS applications. However, changes in the competitive market place as well as legal and regulatory requirements have caused many banks to consider application packages from vendors such as SAP AG. These packages can provide quick to implement and less expensive applications that support new competitive environments as well as assured compliance with new regulations and can provide more integration with other business processes.

However, banks want to have the confidence that the application, and the platforms it runs on, can support their business volumes. This is especially true for large banks. The purpose of the measurements shown and discussed in this report is to help demonstrate large core banking business volumes. In particular, we concentrated on the Database Server (DB Server) and DB2 10 for z/OS [1, 2, 3, 4], as they are key to an SAP solution-based system's scalability and availability.

SAP's Banking Services (SBS), and its core banking component, are part of the SAP for Banking portfolio. For an overview and more information of the entire solution see [5]. For many large banking customers, core banking is the most resource intensive part of the solution. This is because it often has critical windows within which all the processing must be completed. For example, some customers want to complete all the account balancing activity at month end in one night. Often banks also have near continuous availability and high security requirements. To discuss and help satisfy these and other banking needs, SAP has co-authored a paper with IBM, "*SAP for Banking on System z Reference Architecture*" [6].

We have a long history of working with SAP on core banking. For example, there were a set of core banking measurements done in 2002 with DB2 V6 for z/OS and a 75,000-account database. Since then we have tested newer versions of the application with newer versions of DB2, and expanded the number of loaded accounts to 5 million and later 40 million. We have over the years published on the web technical papers discussing some of these as well as other experiences with this workload. For example, there are currently four technical papers available on the World-Wide Web [7, 8, 9, 10].

This paper describes a set of measurements we did with 150 million loaded accounts, SAP AG's most recent SBS 7.0, our most recent DB2® 10 for z/OS, our largest current zEnterprise® system [11], and our System Storage® DS8800®, the most advanced model in our high-end disk portfolio [12]. Our goal was, within schedule and other resource limits, to reach significantly higher business transaction processing rates in the more stressful 150 million-account environment using SAP's latest Banking release.

2 Reminders

Neither this document nor any part of it may be copied or reproduced in any form or by any means or translated into another language without the prior consent of the IBM Corporation. IBM makes no warranties or representations with respect to the content hereof and specifically disclaims any implied warranties of merchantability or fitness of any particular purpose. IBM assumes no responsibility for any errors that may appear in this document. The information contained in this document is subject to change without any notice. IBM reserves the right to make any such changes without obligation to notify any person of such revision or changes. IBM makes no commitment to keep the information contained herein up to date.

The results shown are based on specific workloads run in a controlled environment. The actual throughput that any user will experience will vary considerably from these results. Therefore, no assurance can be given that an individual user will achieve throughput equivalent to the performance stated here.

All customer examples cited or described in this paper are presented as illustrations of the manner in which some customers have used IBM products and the results they may have achieved. Actual environmental costs and performance characteristics will vary depending on individual customer configurations and conditions.

This publication was produced in the United States. IBM may not offer the products, services or features discussed in this document in other countries, and the information may be subject to change without notice. Consult your local IBM business contact for information on the product or services available in your area.

Information about non-IBM products is obtained from the manufacturers of those products or their published announcements. IBM has not tested those products and cannot confirm the 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.

3 Trademarks

IBM, the IBM logo, and ibm.com® are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both. If these and other IBM trademarked terms are marked on their first occurrence in this information with a trademark symbol (® or ™), these symbols indicate US registered or common law trademarks owned by IBM at the time this information was published. Such trademarks may also be registered or common law trademarks in other countries. A current list of IBM trademarks is available on the web at “Copyright and trademark information” at www.ibm.com/legal/us/en/copytrade.shtml.

SAP, SAP R/3, SAP ERP, mySAP, mySAP.com, xApps, xApp, ABAP, BAPI, SAP NetWeaver and all SAP product and service names mentioned herein are trademarks or registered trademarks of SAP AG in Germany and in several other countries all over the world.

4 Feedback

Please send comments or suggestions for changes to msheets@us.ibm.com

5 Acknowledgements

This paper is dedicated to Howard E. Poole in recognition of his more than four decades of leadership, inspiration, and contributions to IBM Corp. and his co-workers.

The authors would also like to recognize the contributions of Heiko Gerwens of SAP AG, Paul Delessio, Andrea Fuga, Akiko Hoshikawa, Lee LaFrese, Dave Levish, Veng Ly, and Akira Shibamiya, and Steve Turnbaugh, all of IBM.

6 Workload and Measurement Background

As part of IBM's continuous testing and measuring of its products, we have been testing and measuring SAP's core banking application for several years. We do this to gain experience with the application as it is used by customers and as it evolves. In this paper, we focused on two business scenarios. The first is Posting. This involves transactions moving money into and out of accounts, displays, and some on-line activities. The second business scenario is Balancing. This involves periodic account balancing – such as monthly interest and fees calculation. Both of these are discussed further below.

We performed the same business functions during these tests as SAP's TRBK Standard Application Benchmark. However, the measurements we did were stress tests – not SAP certified benchmarks. One of the key aspects of this set of measurements was that is was done with a one hundred and fifty million loaded and active accounts database. This is about the size of many large banks. Additionally, while a major goal of these measurements was to get as much throughput as possible in the larger configurations, we did not resort to "benchmark specials". For example, we used the same buffer pool and coupling facility parameters for all the measurements and both workloads. Basically all we varied was the load and number of processor brought to bear on the load.

From our previous experiences with these workloads, we anticipated that we would be stressing many parts of the hardware and software. This is particularly true with respect to database I/O. We used our top of the line zEnterprise z196 because, as a balanced system, it had the highest I/O bandwidth of any System z. In addition, we used FICON Express8 connectivity to our storage subsystems. At the time the measurements began, they were the highest capacity fastest links available on System z. Finally, we used five IBM Storage Subsystem DS8800s. This was the first storage subsystem to exploit the 8 gigabits per second (Gb/s) FICON Express8 capability as well as improved performance with faster controllers, host adapters, device adapters, 2.5-inch, 6 Gb/s SAS (serial-attached SCSI) drives, and latest SSD technology. These all help to deliver a dramatic boost in data throughput in a more compact footprint—a powerful combination aimed at taming the most demanding enterprise applications - like core banking.

6.1 Posting Workload

A posting is a deposit or withdrawal from a customer's account. Typical examples of a posting are a payment out of the account (such as a check you write to pay a bill) or a deposit into your account (such as your net pay). Postings can occur anytime, day or night, especially as more Internet banking takes place. However, most postings take place during first shift. While they can occur virtually every day, they are most frequent on business days.

This workload was developed by SAP to try to simulate a customer environment. The workload consists of interactive "users" going through repetitive cycles of 15 dialogue steps. The vast majority of the postings, about 86%, are mass postings in the first dialogue step through an adaptor via the SAP Business Application Programming Interface (BAPI®). The intent is to reflect that most postings occur through electronic data interchange (EDI) from other financial institutions. For example, one bank sending a group of deposits and withdrawals to another bank. There are other individual payments posted as well as several other types of transactions done via dialogue processing as shown in Table 1 below. In one cycle through the workload steps there are a total 175 postings are done. The whole cycle uses many accounts to avoid creating “hot spots” in the data. The key metric of throughput is postings per unit time.

Step	Operation
1	Create a total of 150 postings via five BAPI calls
2	Create five postings
3	Create bank statement
4	Read postings for account
5	Read details of postings
6	Create five postings
7	Create one bank statement for account
8	Create five postings
9	Create one bank statement for account
10	Create five payment orders
11	Read balances of account
12	Create five postings
13	Create one bank statement for account
14	Read balances for account
15	Read master data for account

Table 1: Posting Workload Steps

6.2 Balancing Workload

Account balancing is performed periodically for each account to calculate charges (fees) and interest as well as posting these. It is usually a series of batch jobs executed at night once a month or quarter depending on the institution and the type of account. Usually account balancing is done for all accounts as soon as possible after the period close such as month-end. It usually is not spread throughout the period.

This workload was developed by SAP also. All the accounts have identical attributes and characteristics. Before doing the account balance, each account has 20 items posted on 20 workdays. The key metric of throughput is accounts balanced per unit time.

7 Configurations

7.1 Hardware Environment

System z DB Server: Measurements were performed on a single 4-book z196 M80 with 80 CPs and 655GB memory. We configured two z/OS LPARs each with 30 CPs and 192GB memory. Additionally we had two CFs each with eight ICFs and 64GB memory connected via eight virtual links using CF level 17.

Storage: For the database we had four IBM System Storage Servers DS8800 Dual Frame 2421-951/95E - each with eight ranks of 300 GB Solid State Disks (SSDs), 32 ranks of 15K-RPM 146GB Hard Disk Drives (HDDs), sixteen FICON Express8 adapters, and 384GB cache including 12GB NVS cache. Our log DASD was a DS8800 Single Frame 2421-951 with 16 ranks of 15K-RPM 146GB HDDs, eight FICON Express8 adapters, and 128GB cache including 4GB NVS cache.

For these measurements, we used approximately 37 TB disk space for the DB2 subsystems including DB2 catalog/directory, active logs, DB2 sort work files, SAP Basis components, core Banking database, DFSORT work files, and temporary files for DB2 utilities. The allocated disk space for the core SBS database is 12.5 TB (Tablespaces: 6.4 TB and Indexspaces: 6.1 TB). All of this was assigned to SSDs. The actual used data is less than the allocated.

Application Servers: We used up to 65 POWER7 model PS701 blades populating five BladeCenters as well as a Presentation Server running on a p7 740 model 8205-E6B with 16-core 3.55 GHz SMT4 with 256GB memory. The PS701 blades each had eight cores running 3.0 GHz, SMT4, and 128GB memory. One of the blades ran a stand-alone ENQ without replication. The rest of the blades were for regular dialog and batch processing. We utilized these blade application servers in two configurations: BladeCenter and System z BladeCenter Extensions (zBX). However, because of schedules, we never had the opportunity to use the zEnterprise Unified Resource Manager in the zBX environment or the intraensemble data network (IEDN).

Network: 10 Gigabit Ethernet networks were used for all connections. The application servers were connected via a 10 Gigabit Ethernet switch to the z196 via eight short range (SR) ports using Open System Adapters (OSA) OSA-Express3 adapters.

Following are conceptual views of the configurations.

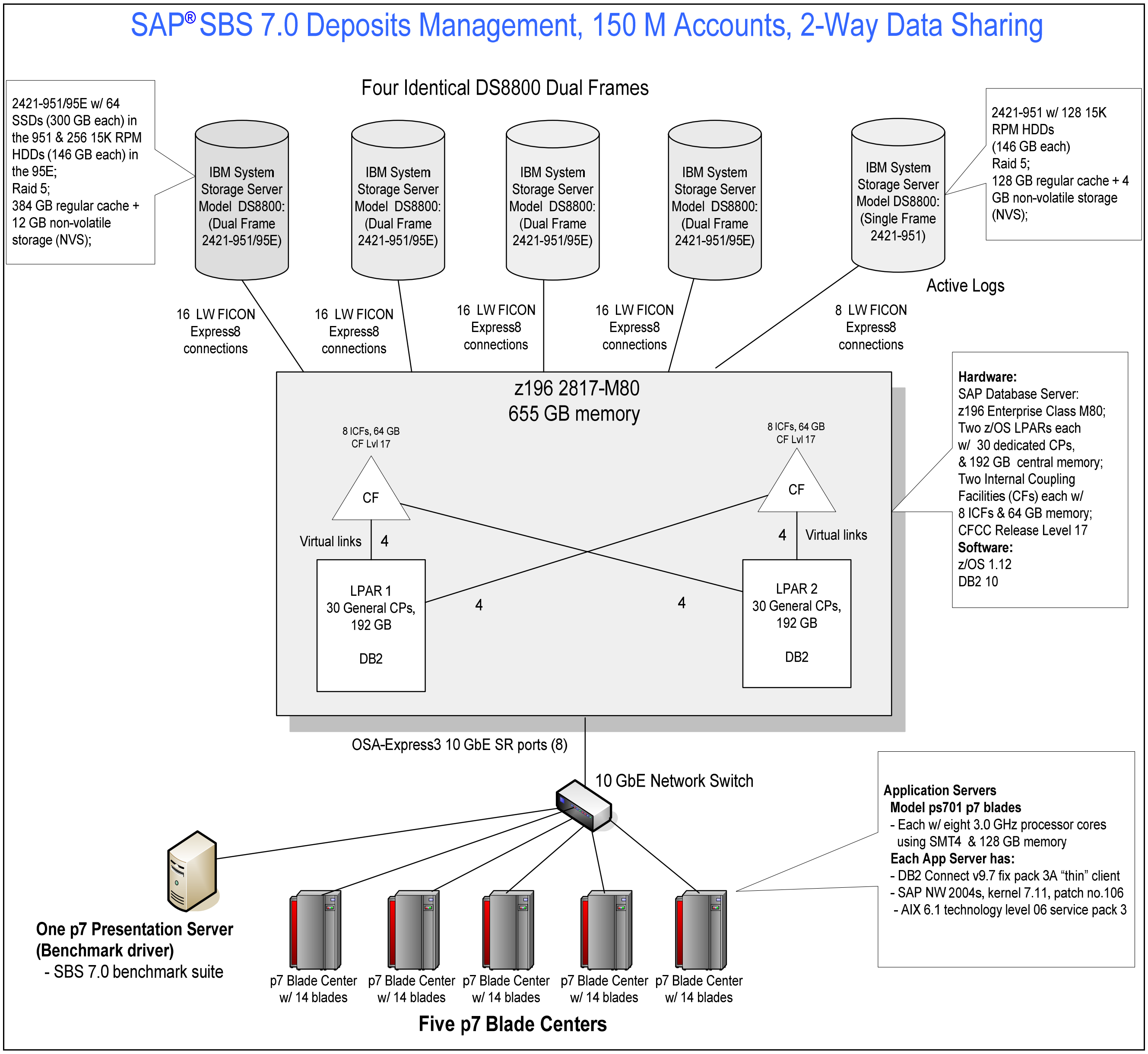

Figure 1: Hardware Conceptual View - Five BladeCenters

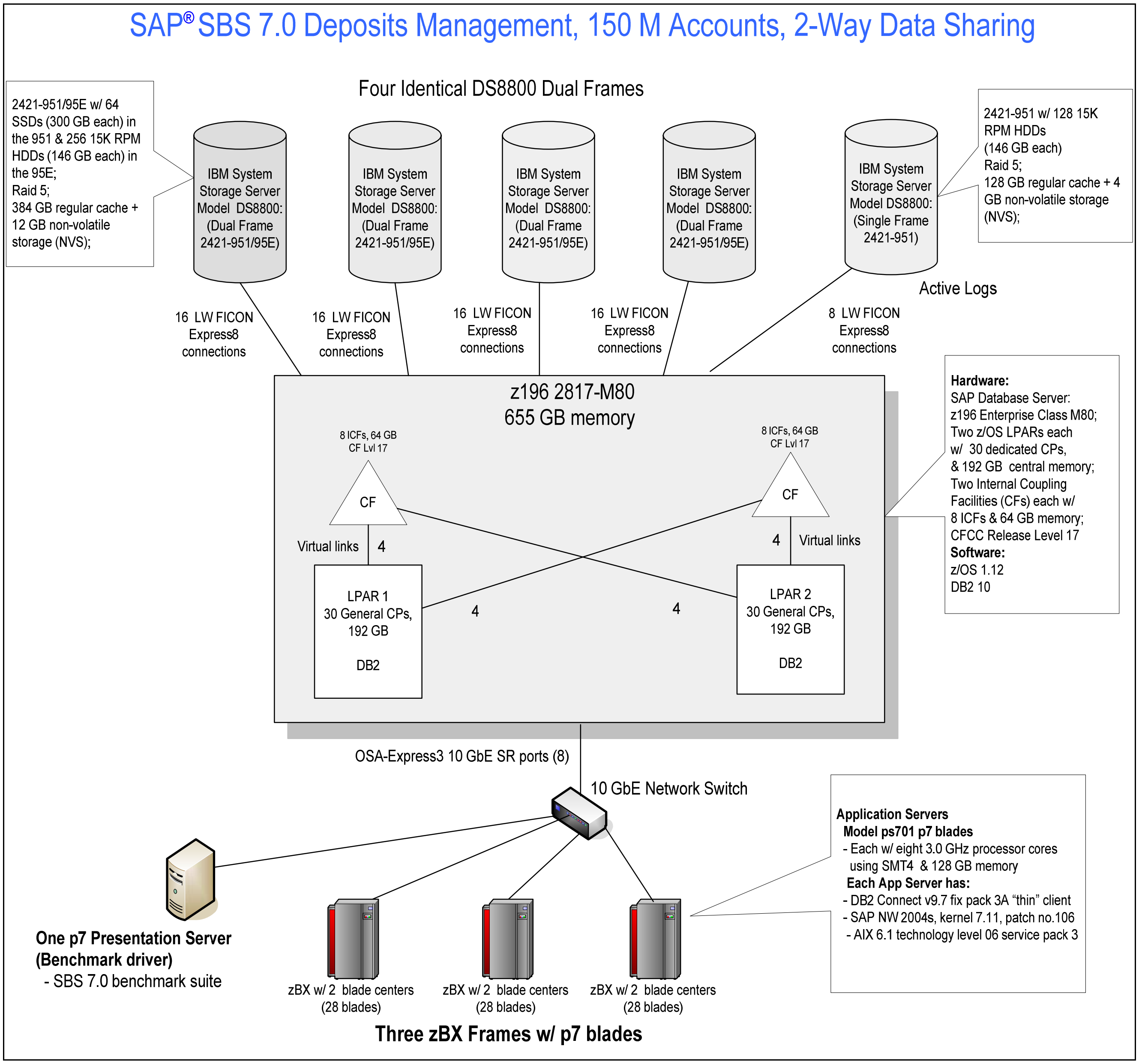

Figure 2: Hardware Conceptual View - zBX

7.2 Software Environment

z/OS

z/OS release 01.12.00 (R1.12)
DB2V10 GA dated October 2010
IBM DB2 Connect "Thin Client": DB2 v9.7.0.3, Fix Pack 3a

AIX

AIX 6.1

oslevel -s
6100-06-03-1048

SAP Application Levels
SAP NetWeaver 7.0 EHP1

Software Component	Release	Level	Support Package	Description
SAP_ABA	711	0007	SAPKA71107	Cross-Application Component
SAP_BASIS	711	0007	SAPKB71107	SAP Basis Component
PI_BASIS	711	0007	SAPK-71107INPIBASIS	Basis Plug-In
ST-PI	2008_1_710	0004	SAPKITLRE4	SAP Solution Tools Plug-In
SAP_BW	711	0007	SAPKW71107	SAP Business Warehouse
FINBASIS	700	0015	SAPK-70015INFINBASIS	Financial Basis
SEM-BW	700	0015	SAPK-70015INSEMBW	SEM-BW 700: Add-On Installation
FSAPPL	300	0008	SAPK-30008INFSAPPL	SAP Banking Services

Table 2: SAP Application Levels

DB2 10 for z/OS
SBS 7.0, like previous versions of the application, utilizes Unicode. DB2 for z/OS uses a fully functional UTF-16 implementation of Unicode. As in the past, we exploited System z's unique hardware data compression and DB2 compression dictionaries to provide the most benefit for the least cost.

8 Measurement Results

While a major goal of these measurements was to get as much throughput as possible in the larger configurations in a limited period of time, we did not resort to "benchmark specials". For example, we used the same buffer pool and coupling facility parameters for all the measurements and both workloads. During the course of this effort many tests and measurements were performed and data collected. They varied from familiarization, debugging, and tuning to "golden" runs. It is beyond the scope of this paper to show them all. Basically, all we varied was the load and number of processor brought to bear on the load.

The measurement results listed below were selected as the most useful. The measurements will be discussed more in Section 9, "Analysis", on page 15.

8.1 Posting Workload

Below are the selected measurement results.

Configuration Short Name	2WDS 2CP	2WDS 4CP	2WDS 16CP	2WDS 32CP	2WDS 48CP	2WDS 60CP
Runid	S10722V4	S10722V3	S10720V1	S10718V2	S10718V1	S10624V2
Processor	2817-M80	2817-M80	2817-M80	2817-M80	2817-M80	2817-M80
Data Sharing Members	2w Data Sharing	2w Data Sharing	2w Data Sharing	2w Data Sharing	2w Data Sharing	2w Data Sharing
# of z/OS LPAR	2	2	2	2	2	2
z/OS LPAR size	1 cp	2 cps	8 cps	16 cps	24 cps	30 cps
Accounts	150 Million	150 Million	150 Million	150 Million	150 Million	150 Million
ETR						
dialog steps/sec	93.53	191.03	581.64	1,007.80	1,295.35	1,408.24
postings/hr	3,928,260	8,023,260	24,428,880	42,327,600	54,404,700	59,146,080
DB ITR						
dialog steps/sec	112.24	219.90	675.23	1,154.48	1,516.71	1,664.78
postings/hr	4,714,101	9,235,939	28,359,508	48,488,000	63,702,008	69,920,889
Avg. z/OS CPU Utilization	83.33%	86.87%	86.14%	87.30%	85.41%	84.59%
Avg. CF1 Utilization	2.0%	3.2%	8.6%	16.6%	22.6%	23.2%
Avg. CF2 Utilization	1.6%	7.4%	12.0%	23.6%	32.9%	35.5%
Avg. diag. response time (sec)	2.524	1.027	0.854	1.126	1.565	2.134
I/O Interrupt rate	17,796	34,606	115,991	188,269	235,000	248,000
Database I/O rate	7,728	13,949	54,307	94,363	124,267	138,647
Avg. DB2 Log rate						
I/O per sec	1,297	3,301	4,743	7,361	8,082	8,296
MB/sec	5.82	17.32	26.36	44.38	57.36	62.90
DBM1 storage below 2G from DB2PM (MB)	76	56	83	124	155	164
Avg. Application Servers (36) Utilization.						55%
Avg. ENQ Application Server Utilization						43%
# of SAP DV instances	10	18	54	96	128	144
Total # of diag. wp's	80	144	432	768	1024	1152

Table 3: Posting Workload Results

8.2 Balancing Workload

Below are the selected measurement results.

Configuration Short Name	2WDS 48CP
Runid	S10712V1
Processor	2817-M80
Data Sharing Members	2w Data Sharing
# of z/OS LPAR	2
z/OS LPAR size	24 cps
Accounts	150 Million
Elapsed time (seconds)	
avg	14,436
max	14,524
min	14,366
Elapsed time (minutes)	
avg	240.60
max	242.07
min	239.43
ETR	
accts/sec	10,327.73
accts/hr	37,179,840.26
postings/hr	113,398,512.81
DB ITR	
accts/sec	12,316.92
accts/hr	44,340,894.77
postings/hr	135,239,729.05
Avg. z/OS CPU utilization	83.85%
I/O Interrupt rate	106,113
Database I/O rate	21,459
Peak DB2 Log rate	
I/O per sec	5,573
MB/sec	410.40
DBM1 storage below 2G	190M
Avg. Application Servers (64) Utilization.	24%
Avg. ENQ Application Server Utilization	11%
# of SAP batch instances	128
# of jobs per SAP inst.	6
Total # of jobs	768

Table 4: Balancing Workload Results

9 Analysis

Once again, while a major goal of these measurements was to get as much throughput as possible in the larger configurations in a limited period of time, we did not resort to "benchmark specials". We used a 150 million-account database for these measurements. Past experience (e.g., page 17 of [6]) shows that the costs per business transaction are a function of the total number of accounts in the database. In the Balancing measurements, all 150 million accounts were balanced to approximate a fully operational large bank.

This section discusses some aspects of these measurements common to both workloads. There are also some comparisons made between the two workloads. We have to remember, however, that these two workloads do completely different business processes and are not interchangeable.

Analysis of various aspects of these measurements was done. In some cases, this analysis used standard Large Systems Performance Reference (LSPR) techniques and methodology [13]. It is assumed the reader is familiar with the terms therein, especially External Throughput Rate (ETR), Internal Throughput Rate (ITR) and the corresponding ratios (ETRR and ITRR). All the ITR and ITRR based results were considering the DB Server utilizations only. Thus, they should be considered as DB ITRs or DB ITRRs.

SAP has several infrastructure services such as enqueue (ENQ), message (MSG), gateway (GW) and spool (SPO). There are several options on how and where to run these. However, ENQ processing is frequently used and can play a role in overall throughput. To help achieve better results, we ran all these functions on a dedicated PS701 blade. Further, we used two SAP instances on that blade. One ran ENQ and MSG (or ASCS/SCS – also sometimes called a stand-alone ENQ). The other instance ran GW and SPO. This allowed the ENQ faster access to processor resources. We did not use ENQ replication.

As described in Section 7.1, "Hardware Environment" on page 9, we had four identical DS8800 Dual Frame Storage systems for the database. We also had a separate DS8800 for the active logs. We put the entire database (12.5 TB) on SSDs. This was necessitated because of the scale of the measurements and the nature of the workloads.

While the business functions of the two workloads are not at all similar, both these workloads had heavy I/O requirements – but in different and complementary areas. A comparison is shown graphically in Figure 3: I/O Characterization by Workload, below. Posting has much higher database I/O rates than Balancing. Posting has more random I/O and behaves more like a typical OLTP workload. Conversely, Balancing heavily exploits sequential pre-fetch and, as a result, moves more data per I/O operation. Balancing has much higher logging data rates (see the right axis) because it is more write intensive than Posting. Balancing benefited from having a dedicated logging storage system, its associated I/O paths, as well as DB2 log striping. We used four log stripes. We have seen these I/O characteristics of the different SBS workloads for many years.

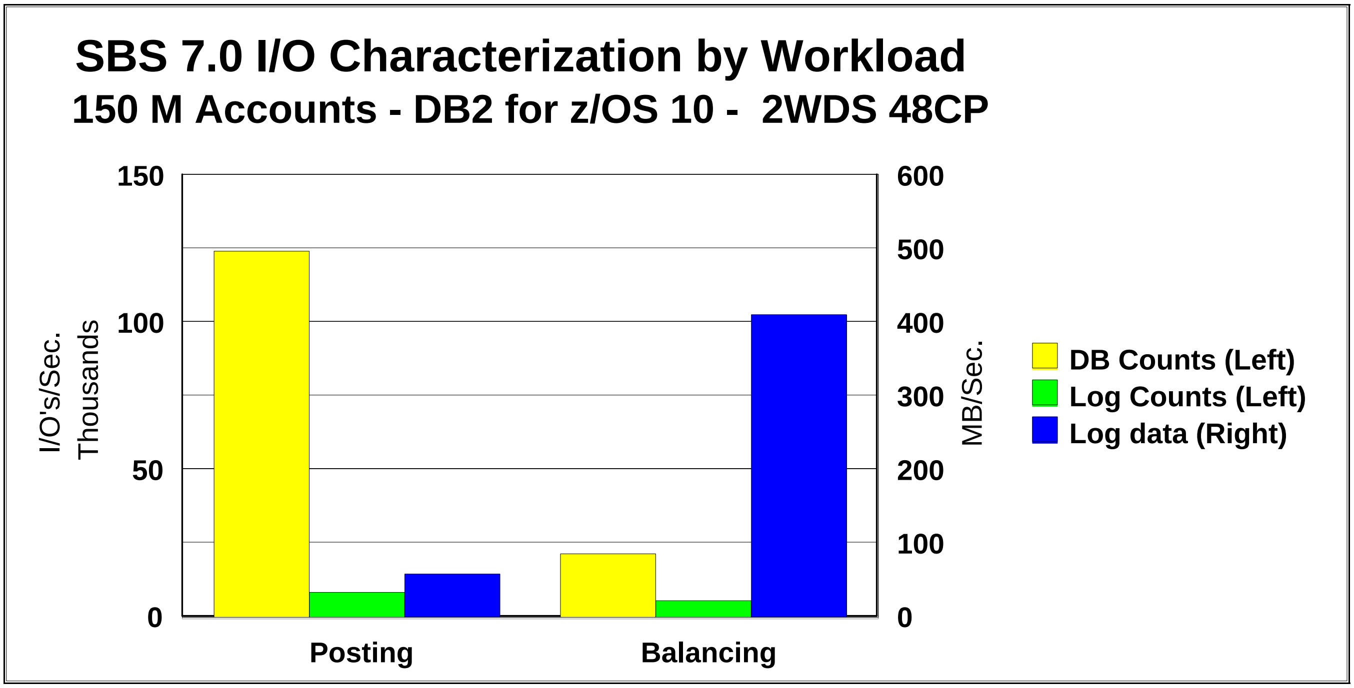

Figure 3: I/O Characterization by Workload

DB2 10 for z/OS

In the time that passed since our previous measurements with these workloads, a lot has changed. For example, the hardware (processor and storage systems), and software, especially DB2, have made significant advances. There is strong synergy between the hardware, operating system, and database. Examples are DB2's exploitation of 1 MB page frames, long-term page fix, and special optimizations for more recent System z machines. All these contributed to the high throughput and good scalability of the measurements.

As well, DB2 10 made significant improvements in its scalability - supporting large numbers of concurrent operations. Items that were particularly important for these workloads were the below the 2 GB bar virtual storage constraint relief, extended exploitation of 64-bit virtual storage, EDM pool and buffer manager serialization improvements, and latch contention improvements. For example, the latch contention improvements directly help logging for both Posting and Balancing workloads.

zHPF

High Performance FICON for System z (zHPF) can significantly improve I/O capacities of FICON channels when performing small block (e.g., 4KB) transfers. This is because zHPF reduces the resources required of channel processors, control unit ports, and links [14]. DB2 exploits zHPF through z/OS's Media Manager. Over the years, zHPF and its effectiveness has been improved through items like larger byte limits, multi-track data transfers, and improved channels (e.g., FICON Express 8 and FICON Express8S). Both Posting and Balancing workloads were excellent exploiters of zHPF.

zIIP

The System z Integrated Information Processor (zIIP) has been quite popular with customers in the more than four years since it was introduced. Some of its earliest and biggest exploiters were SAP for System z DB2 for z/OS database servers. We had published in [15] several zIIP measurements. Over the years, there have been some changes in the use of zIIPs – including further exploitation.

To avoid system reconfigurations and thereby improve our measurement productivity, our z196 was configured without zIIPs. However, we are able to determine the zIIP exploitation from the PROJECTCPU field in RMF's workload activity report. We found that the SBS 7.0 workloads continue to be excellent zIIP exploiters. This can represent a significant customer cost savings as work redirected to zIIPs does not contribute to MSU's consumed and zIIP processors also have a lower cost.

Data-Sharing

Our goal with the Parallel Sysplex Data Sharing configuration was primarily high throughput and, to a lesser extent, scalability. While this configuration did have very good availability characteristics, its availability could have been further improved by eliminating some single points of failure and other configuration changes. For example, we could have run a replicated ENQ, used multiple server footprints, as well as an external coupling facility.

With respect to the coupling facilities, we can see that posting placed a higher load on the coupling facilities than balancing. For example, a comparison of the two-way data-sharing 48 CP measurements (short name 2WDS 48CP) shows that the Posting CFs were 23% and 33% busy while the Balancing measurement shows both CFs averaged less than 2% busy. Notwithstanding, neither workload was constrained by the coupling facilities or their links. Posting had higher cross system interest than Balancing. Part of the reason for this is the Balancing application was designed with data-sharing in mind. See Section 9.2, "Balancing Workload" on page 20 for more information.

DB Growth

We analyzed the database growth during these measurements. The results for the DB2 tables that contributed the most to DB growth are listed in the sections that follow. These results should only be considered a starting estimate of DB growth. The actual growth is highly dependant on the specific application implementation. As well, the actual physical DB growth will depend on the compression options chosen. We chose data compression only, which exploits unique System z hardware and DB2 directories. Installations should consider these factors, their requirements for keeping data on-line, and their archiving strategies when planning an implementation.

9.1 Posting Workload

Our high water result of 59.1 million postings per hour significantly eclipsed our previous 15.7 million postings per hour. This was accomplished in the more difficult 150 million accounts with SBS 7.0 environment while the earlier result was with 40 million accounts and a less functionally rich TRBK4 SAP banking level. It is even more impressive that the below-the-bar (2 GB) virtual storage constraint relief and extended exploitation of 64-bit virtual storage provided in DB2 10 allowed us to reach 59.1 million postings per hour using only two DB2 data-sharing members. The previous 15.7 million result was done with four members. The previous two-member result was 8.4 million postings per hour.

Figure 4, below, graphically shows the throughput and response times we measured for this workload. In our most recent measurements, we were limited somewhat at the high end by inserts into the database. The response time was usually in the one to two second range. The exception is the two-way data-sharing two CP measurement (short name 2WDS 2CP - two one-way processors) measurement where we most likely had queuing for the processor. This is confirmed by the increase in the amount of DBM1 storage below-the-bar compared to the two-way data-sharing four CP (short name 2WDS 4CP - two two-way processors) measurement.

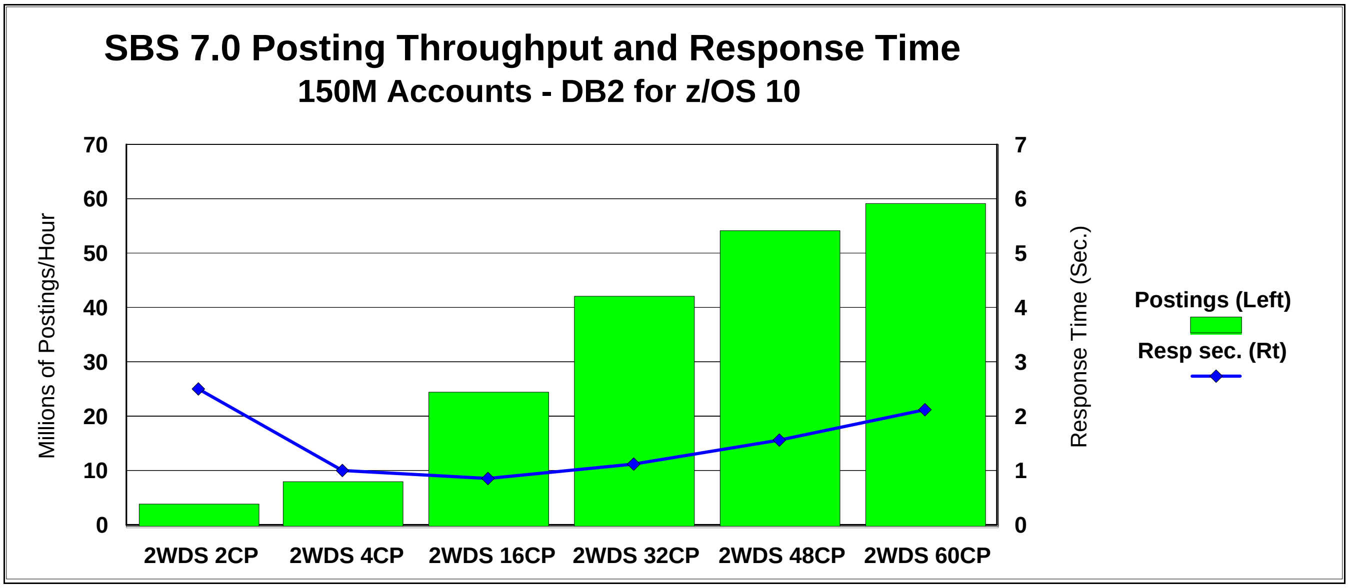

Figure 4: Posting Throughput and Response Time

The DB ITRR metric can be used as an indicator of processor efficiency. Figure 5, below, shows basically the same graph as the throughput (Figure 4, above) because the utilizations of the measurements were very similar to each other.

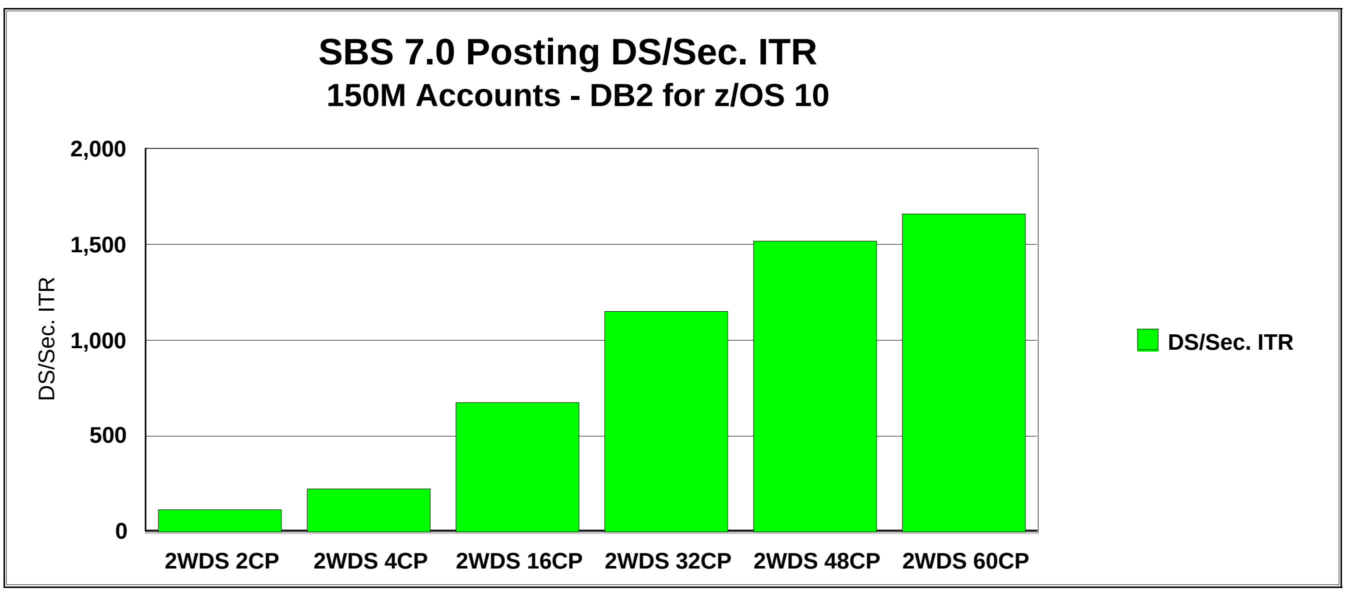

Figure 5: Posting ITR in DS/Second

<u>DB2 10 for z/OS</u>

In general, the amount of DBM1 storage below the 2 GB bar with DB2 10 was much less than we had seen previously with DB2 9. For example, our 59.1 million postings per hour measurement with DB2 10 used 164 MB for each of the two members while the 15.7 million posting measurement with DB2 9 used 788 MB for each of the four members.

zHPF

The posting workload, like the balancing workload is an excellent exploiter of zHPF (see Section 9, "Analysis" on page 15). Log I/O used zHPF 88% of the time. Logging used one of the newer features of zHPF - multi-track support. The database I/O's used zHPF 98% of the time. Because the database I/O activity is dominant, the overall zHPF percentage for this workload was also 98%. This is an improvement from when we first tested SSD's with SAP over two years ago. At that time, we observed the database using zHPF 95% of the time for a modified version of the Posting workload (see page 16 of [7]).

zIIP

To avoid system reconfigurations and thereby improve our measurement productivity, our z196 was configured without zIIPs. However, we are able to determine the zIIP exploitation from the PROJECTCPU field in RMF's workload activity report. We found that the Posting workload could have redirected 60% of the DRDA work or 52% of all the DB2 CPU time (e.g., the sum of DBM1, DB2 Master, DRDA, and IRLM) to zIIPs. This is the equivalent of 47% of the entire LPARs' CPU time to zIIPs. As an example, for this workload we could have configured this system as two LPARs each with 16 regular processors and 14 zIIPs.

Data-Sharing

Figure 6, below, shows the utilization of the CFs while running posting. The utilizations of the CFs differ because different objects, with their application defined unique inter-system interest, were allocated to different CFs. CF2 scales with throughput because it contained the lock structures while CF1 contained the group buffer pools. Prior measurements (see page 14 of [7]) indicate a coupling efficiency of 83%.

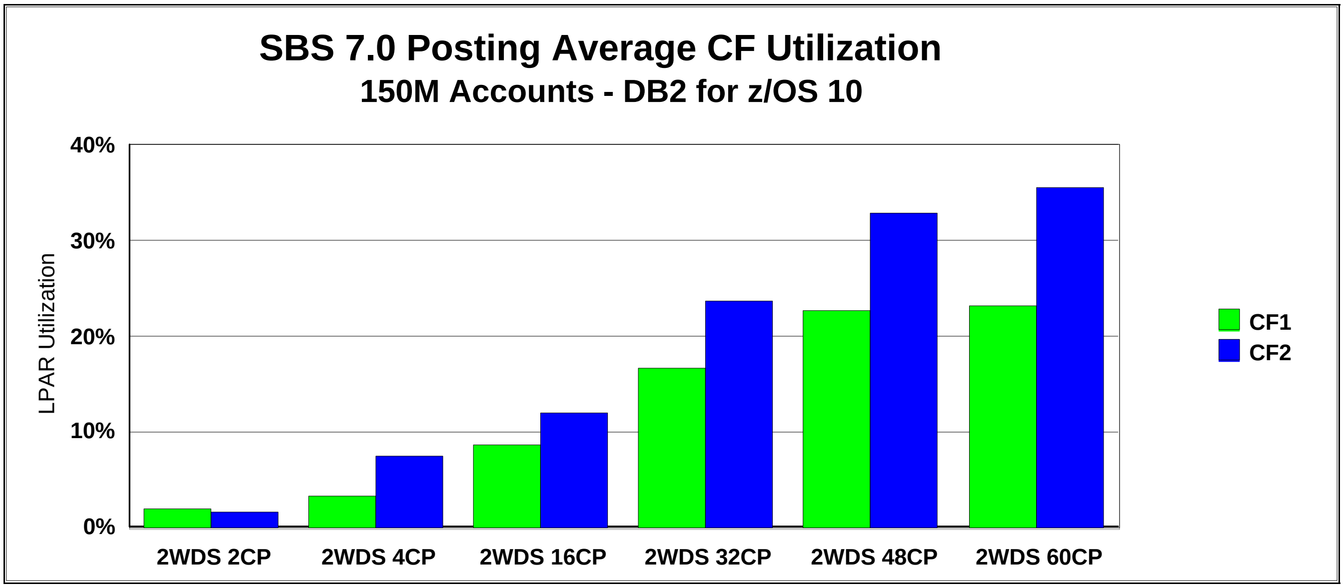

Figure 6: Posting Average CF Utilization

DB Growth

Listed in Table 5, below, are the DB2 tables that contributed the most to DB growth. The five tables shown totaled 1,246 MB growth per million postings. Keep in mind that, since we used hardware data compression, the table growth shown is in terms of compressed space. The PAGESAVE parameter approximates the percentage of space saved by data compression. The high numbers shown are typical of many SAP tables. We did not use index compression.

DB2 Table	PAGESAVE	Table Growth (MB / Million Postings)	Index Growth (MB / Million Postings)	Total Growth (MB / Million Postings)
BCA_BCAS_EVBST	75%	8.7	9.8	18.5
BCA_COUNTER	69%	15.0	33.4	48.4
BCA_GL_PAYMITEM	78%	103.0	176.0	279.0
BCA_PAYMITEM	72%	566.0	278.0	844.0
BCA_TRANSFIG	89%	12.9	43.3	56.1

Table 5: Posting Tables That Contribute to DB Growth

9.2 Balancing Workload

With just two DB2 members, our high water result was 37.2 million accounts balanced per hour. As discussed below, this workload encountered logical logging constraints. Therefore, the number of DB2 members has a major effect on this workload. The results with the new DB2 10 significantly eclipsed our previous 10.7 million accounts per hour – even through the previous result had the advantage of having four DB2 members. The previous results also had advantages of a smaller database (i.e., 40 million accounts) and a less functionally rich TRBK 4 SAP banking level. Taking all this into account, the new results were very impressive.

As mentioned earlier, the balancing workload is batch oriented, exploits sequential pre-fetch, and does a great deal of logging. Figure 7, below, demonstrates how well balanced this workload was between the two data sharing members. This is apparent in the measurement results' elapsed times. The first batch job finished in 239 minutes. The last job finished in 242 minutes - less than 1% variance from the average. We were able to process all the 150 million accounts in about four hours.

The well-balanced nature was a result of three factors. First, the two data-sharing members were symmetrical. Second, the application servers were homogenous. Finally, the SAP SBS developers have worked with DB2 for z/OS and data-sharing for many years. As a result, the application exploits data-sharing well. This is discussed more in the Data-Sharing portion of this section.

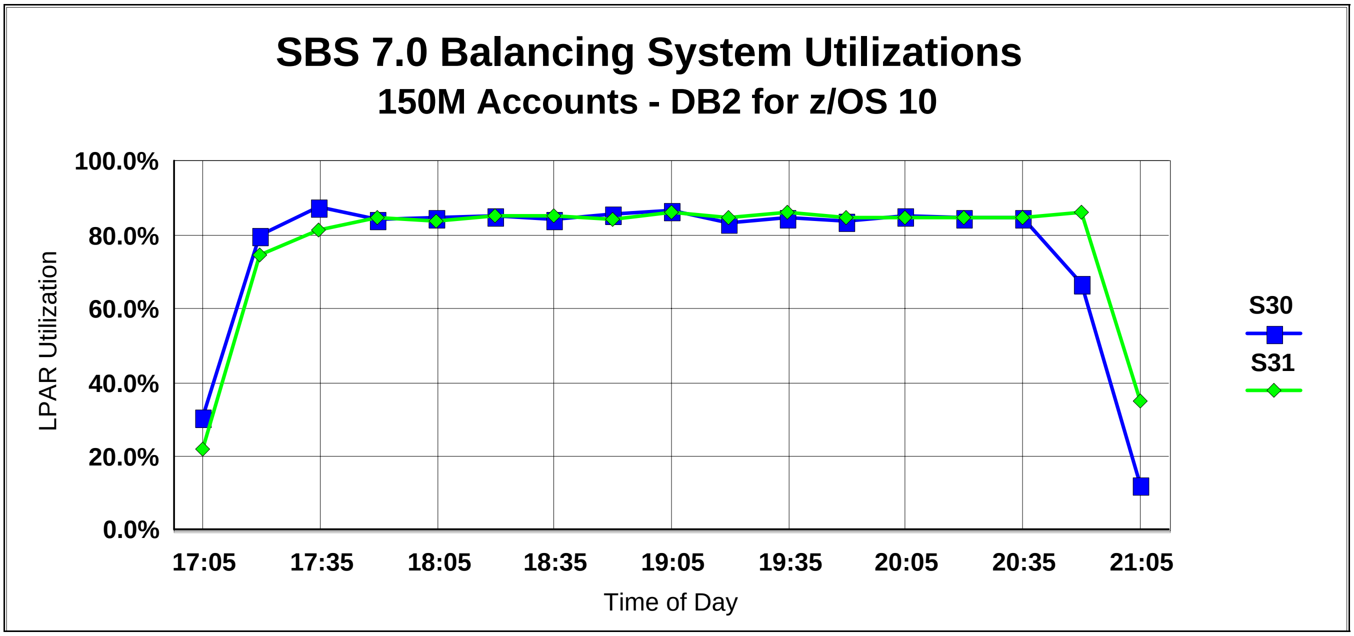

Figure 7: Balancing LPAR Utilizations

DB2 10 for z/OS

DB2 10 made significant advances in the area of logical logging. Previous measurements (see page 12 in [7]) were limited by logical logging to less than six million accounts per hour for two DB2 members. Here we achieved 37.2 million accounts per hour. However, we were still limited somewhat by DB2 logical logging. More DB2 members would alleviate this.

zHPF

The balancing workload, like the posting workload, is an excellent exploiter of zHPF (see Section 9, "Analysis" on page 15). Log I/O used zHPF 100% of the time (rounded to the nearest percent). Logging used one of the newer features of zHPF - multi-track support. The database I/O's used zHPF 92% of the time. Overall zHPF percentage for this workload was 94%.

zIIP

To avoid system reconfigurations and thereby improve our measurement productivity, our z196 was configured without zIIPs. However, we are able to determine the zIIP exploitation from the

PROJECTCPU field in RMF"s workload activity report. We found that the Balancing workload could have redirected 60% of the DRDA work or 58% of all the DB2 CPU time (e.g., the sum of DBM1, DB2 Master, DRDA, and IRLM) to zIIPs. This is the equivalent of 51% of the entire LPARs' CPU time to zIIPs. As an example, for this workload we could have configured this system as two LPARs each with 15 regular processors and 15 zIIPs. This is somewhat higher than the Posting workload because DB2 10 now redirects buffer manager pre-fetch and deferred write processing to zIIPs.

Data-Sharing

To drill down into data-sharing some more, each SAP instance is assigned to a specific application server. In turn, during normal circumstances, each application server is assigned to a specific data-sharing member. In addition, the associated tables have the same, or more, partitions than there are instances. As the jobs in each instance run through their packages, the result is that each data-sharing member has affinity to its set of adjacent accounts [16]. This minimizes group interest. This is further demonstrated by the low CF utilization in Figure 8, below. From prior measurements (see page 12 of [7]), we estimate the coupling efficiency to be 92%.

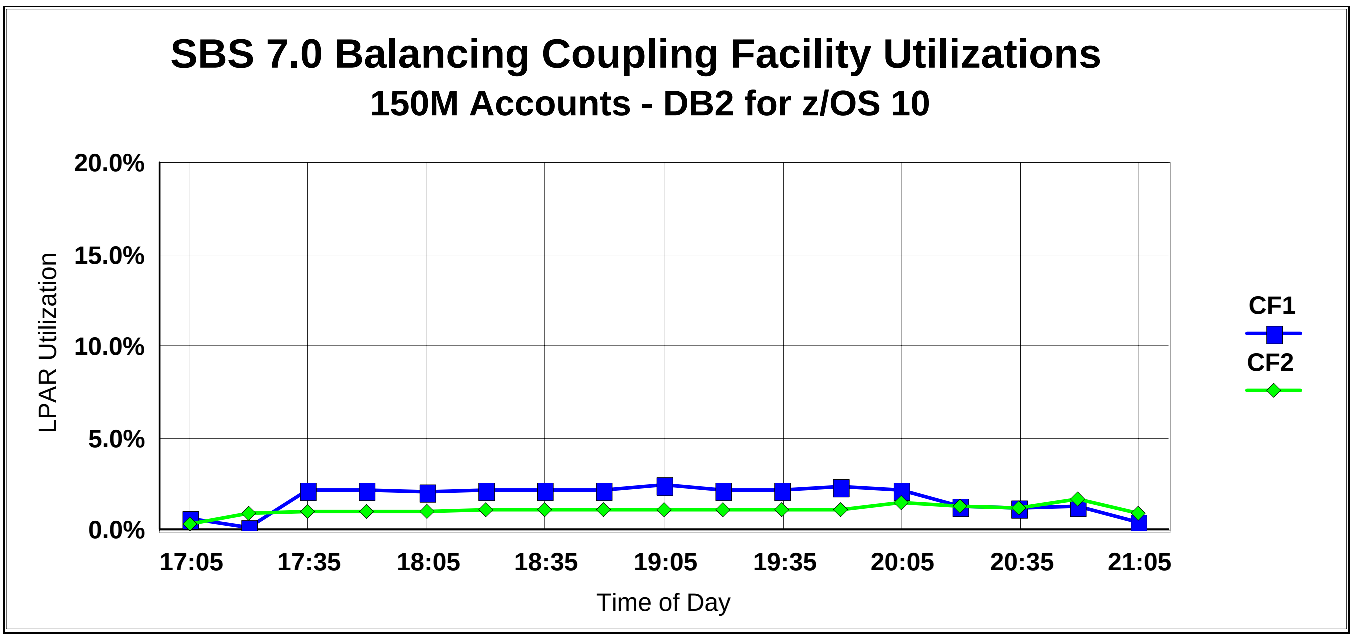

Figure 8: Balancing Coupling Facility Utilizations

DB Growth

Listed in Table 6, below, are the DB2 tables that contributed the most to DB growth. The eight tables shown totaled 15,821 MB per million accounts balanced. Keep in mind that, since we used hardware data compression, the table growth shown is in terms of compressed space. The PAGESAVE parameter approximates the percentage of space saved by data compression. The high numbers shown are typical of many SAP tables. We did not use index compression.

DB2 Table	PAGESAVE	Table Growth (MB / Million Accounts)	Index Growth (MB / Million Accounts)	Total Growth (MB / Million Accounts)
BCA_GL_PAYMITEM	78%	314.2	536.8	851.0
BCA_PAYMITEM	72%	1,726.3	847.9	2,574.2
BCA_TRANSFIG	89%	30.0	101.0	131.0
BCA92	81%	87.0	126.0	213.0
BKK92_POSTINGS	75%	70.0	256.0	326.0
BKK92_SUMS	74%	375.2	295.9	671.0
BCA96	84%	2,050.0	2750.0	4,800.0
BCA92_RESTART	81%	1,765.0	4490.0	6,255.0

Table 6: Balancing Tables That Contribute to DB Growth

10 Conclusions

From a high-level summary perspective, these results show:

- the superior performance and scalability of IBM's hardware and database software when running the SAP for Banking solution, and emphasize IBM's leading role in the IT environment for banking.
- this world-class scalability and performance are some of the reasons customers choose the IBM System z platform to run their core banking applications.
- the latest generation of the IBM zEnterprise System provides an integrated, hybrid platform with mainframe qualities of service, bringing together the zEnterprise mainframe servers with select IBM blades.
- the strength of IBM System Storage and its flexibility to meet extreme dynamic and varying I/O demands.
- a proven infrastructure reference architecture for banking defined by SAP and IBM for System z and DB2 for z/OS

These measurements represent a significant leap in throughput for both workloads from our previous experience. This is shown graphically in Figure 9 and Figure 10, below. Our new high water results were 59.1 million postings per hour and 37.2 million accounts balanced per hour. Our previous best results were 15.7 million postings per hour and 10.7 million accounts per hour respectively. Further, these new results were done in a much more challenging environment that included an impressively large 150 million account database, as well as a new more functionally rich and more complex level of SAP's Banking Services. The new results were accomplished with only two DB2 members – the minimum number recommended for high availability. Both workloads were measured with the same database, hardware, and tuning parameters – like a production customer.

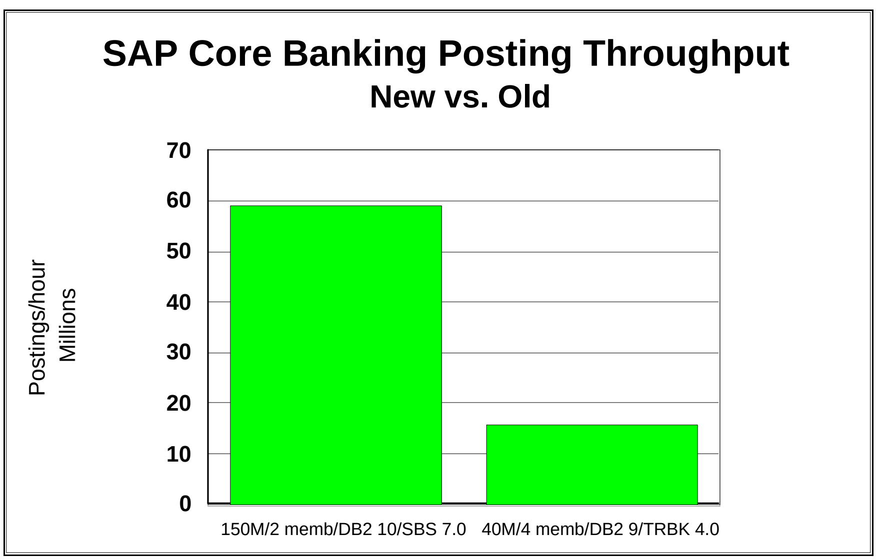

Figure 9: Posting Summary - New vs. Old

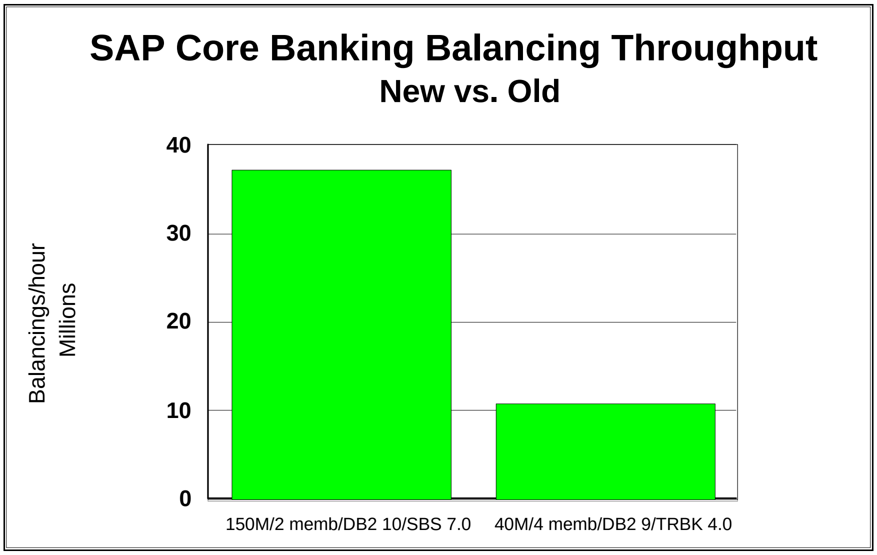

Figure 10: Balancing Summary - New vs. Old

A lot of very impressive hardware was gathered together to help make this happen. Key factors in the success of these measurements were the zEnterprise System, System Storage Server Model DS8800s, and DB2 10 for z/OS. Also contributing were other features such as FICON Express8 with zHPF, OSA-Express3 10 Gigabit Ethernet, SSDs, POWER7 PS701 Application Servers. All these allowed us to provide impressive scalability while providing industrial strength manageability as well as the potential for the highest availability through the use of Parallel Sysplex data-sharing. The scale of these measurements placed very high requirements on the zEnterprise, storage subsystems (both random I/O, and logging I/O), and POWER7 application servers.

The detailed results, as well as analysis, will be used to further improve IBM's products. We want to continue to provide leading edge support for SAP and banking customers.

11 References

[1] IBM Corp. 2010. *IBM United States Software Announcement 210-380, IBM DB2 10 for z/OS improves efficiency and resiliency* http://www.ibm.com/common/ssi/rep_ca/0/897/ENUS210-380/ENUS210-380.PDF

[2] IBM Corp. 2010. *DB2 10 for z/OS* http://www.ibm.com/software/data/db2/zos/

[3] IBM Corp. 2010. *DB2 10 for z/OS Technical Overview* http://www.redbooks.ibm.com/redbooks/pdfs/sg247892.pdf

[4] SAP AG 2010. *SAP on DB2 10 for z/OS – Being More Productive, Reducing Costs and Improving Performance* http://www.sdn.sap.com/irj/scn/index?rid=/library/uuid/005c6b33-aaf0-2d10-fcbb-b42e89ac5791

[5] SAP AG 2011. http://www.sap.com/solutions/industry/banking/ and http://help.sap.com/saphelp_banking463/helpdata/en/e7/c1e7bd0cdf11d28a250000e829fbbd/frameset.htm

[6] SAP AG 2011. *SAP for Banking on System z Reference Architecture* http://www.sdn.sap.com/irj/sdn/db2?rid=/library/uuid/a00e4718-314f-2b10-19a6-a76f257addaf

[7] IBM Corp. 2008. *SAP® Transaction Banking: IBM System z® DB Server Using DB2 V9.1 for z/OS Large Database Measurements* http://www.ibm.com/support/techdocs/atsmastr.nsf/WebIndex/WP101243 or msheets@us.ibm.com

[8] IBM Corp. 2009. IBM System z® *and System Storage DS8000: Accelerating the SAP® Deposits Management Workload With Solid State Disks* http://www.ibm.com/support/techdocs/atsmastr.nsf/WebIndex/WP101442 or msheets@us.ibm.com

[9] IBM Corp. 2010. *SAP Transaction Banking: Case Study Understanding Data Sharing Effects as a Function of Long Distance Coupling Links IBM System z* http://www.ibm.com/support/techdocs/atsmastr.nsf/WebIndex/WP101692 or msheets@us.ibm.com

[10] IBM Corp. 2011. *DB2 10 for z/OS with SAP on IBM System z Performance Report* http://www-.ibm.com/support/techdocs/atsmastr.nsf/WebIndex/WP101845 or msheets@us.ibm.com

[11] IBM Corp. 2010. *IBM zEnterprise System* http://www.ibm.com/systems/z/hardware/zenterprise/

[12] IBM Corp. 2010. *New IBM System Storage DS8800* http://www.ibm.com/systems/storage/news/center/disk/enterprise/

[13] IBM Corp. 2010. *Large Systems Performance Reference* https://www.ibm.com/servers/resourcelink/lib03060.nsf/pages/lsprindexpdf/$file/SC28118714_20100714.pdf

[14] IBM Corp. 2009. *IBM System z10 I/O and High Performance FICON for System z Channel Performance* ftp://ftp.software.ibm.com/common/ssi/sa/wh/n/zsw03127usen/ZSW03127USEN.PDF

[15] IBM Corp. 2006. *IBM System z9 zIIP Measurements: SAP OLTP, BI Batch, SAP BW Query, and DB2 Utility Workloads* http://www.ibm.com/support/techdocs/atsmastr.nsf/WebIndex/WP100836 or msheets@us.ibm.com

[16] SAP AG 2000. *Bank Customer Accounts BCA: Performance Test* http://www.sap.com/belux/industries/banking/pdf/50042191.pdf

12 Appendix – DB2 Buffer Pools

While it is beyond the normal scope of this paper, here is some information on how we defined and used the DB2 buffer pools.

Table 7, below, shows the larger buffer pools, their options, and their group buffer pool sizes. Buffer pool isolation is used extensively for these measurements to ease DB2 performance monitoring and tuning. Highly accessed objects are placed into their own dedicated buffer pools with indexes separated from the corresponding table. Total local buffer pools per DB2 member are about 90 GB and are PAGE fixed in memory to minimize page-steal activity. In general, default settings are used for sequential thresholds, horizontal, and vertical deferred write thresholds. However, DB2 prefetch is disabled (VPSEQT=0) for a few selected objects in their own buffer pools where access pattern is known. Lower horizontal and vertical deferred write thresholds are set for objects, which have high insert/update activities to facilitate smaller more frequent disk writes. This is to minimize the impact of intensive disk write activities that occur during database checkpoints.

Finally, Table 8 on page 30 shows some of the larger more important tables and indexes and which buffer pools there were assigned to

BP name	PGFIX	VP Pages (K)	BP size (MB)	VPSEQT (%)	DWQT (%)	VDWQT (%)	GBpool Size
BP0	YES	100	400	80	5	1	4,000,000
BP1	NO	10	40	100	90	90	20,000
BP2	YES	800	3,200	80	5	1	2,000,000
BP3	YES	600	2,400	80	10	2	3,000,000
BP4	YES	60	240	80	5	1	2,000,000
BP5	YES	800	3,200	80	10	2	4,000,000
BP6	YES	8	32	80	50	10	1,500,000
BP7	YES	6	24	80	50	10	1,500,000
BP8	YES	600	2,400	80	5	1	4,000,000
BP9	YES	1200	4,800	80	50	10	200,000
BP10	YES	600	2,400	80	5	1	4,000,000
BP11	YES	1200	4,800	80	10	2	6,000,000
BP12	YES	150	600	80	50	10	200,000
BP13	YES	300	1,200	80	50	10	200,000
BP18	YES	10	40	80	50	10	20,000
BP19	YES	10	40	80	50	10	200,000
BP20	YES	300	1,200	80	50	10	200,000
BP21	YES	960	3,840	80	50	10	160,000
BP22	YES	50	200	80	50	10	300,000
BP23	YES	20	80	80	50	10	300,000
BP24	YES	150	600	80	50	10	200,000
BP25	YES	240	960	80	50	10	160,000
BP28	YES	240	960	80	5	1	6,000,000
BP29	YES	960	3,840	80	10	2	4,000,000

BP name	PGFIX	VP Pages (K)	BP size (MB)	VPSEQT (%)	DWQT (%)	VDWQT (%)	GBpool Size
BP32	YES	2	8	80	50	10	200,000
BP33	YES	2	8	80	50	10	200,000
BP34	YES	2	8	80	5	1	200,000
BP35	YES	2	8	80	50	10	80,000
BP37	YES	150	600	80	50	10	240,000
BP40	YES	10	40	80	50	10	84,000
BP41	YES	210	840	80	10	2	4,000,000
BP42	YES	390	1,560	0	50	10	2,000,000
BP43	YES	100	400	80	5	1	4,000,000
BP48	YES	1200	4,800	80	5	1	1,000,000
BP49	YES	900	3,600	80	10	2	1,000,000
BP32K	YES	40	1,280	80	5	1	1,250,000
BP32K1	YES	20	640	80	5	1	1,250,000
BP32K2	YES	100	3,200	0	50	30	8,000,000
BP32K3	YES	200	6,400	80	50	10	7,200
BP32K5	YES	32	1,024	0	50	10	50,000
BP32K8	YES	20	640	80	50	10	50,000
BP32K9	YES	150	4,800	80	50	10	50,000
BP8K	YES	240	1,920	80	5	1	6,000,000
BP8K1	YES	720	5,760	80	10	2	6,000,000
BP8K3	YES	120	960	80	10	2	3,000,000
BP8K5	YES	720	5,760	80	10	2	6,000,000
BP8K7	YES	720	5,760	80	10	2	6,000,000
BP8K9	YES	240	1,920	80	10	2	3,000,000
BP16K	YES	2	32	80	50	10	5,120

Table 7: Buffer Pool Definitions

Buffer Pool	Table/Index Space
BP0	BCA_PO_ERR_IDX~ACC
BP2	BCA_BCAS_DUE
BP2	BCA_BCAS_EVBST
BP2	BCA_CN_EV_ACBAL
BP2	BCA_CNSP_ACCT
BP2	BCA_PAYMITEM_NT
BP2	BCA_PO_HD
BP2	BCA_PO_NT
BP2	BCA_PO_ERR_IDX
BP3	BCA_BCAS_DUE Index
BP3	BCA_BCAS_EVBST Indexes
BP3	BCA_CN_EV_ACBAL Indexes
BP3	BCA_PAYMITEM_ENQ Indexes
BP3	BCA_PAYMITEM_NT Index
BP3	BCA_PO_HD Indexes
BP3	BCA_PO_IT Indexes

Buffer Pool	Table/Index Space
BP3	BCA_PO_NT Index
BP3	BCA_PO_ERR_IDX Index
BP4	BCA_GL_PAYMITEM
BP5	BCA_GL_PAYMITEM Index
BP6	BCA92
BP6	BKK92_POSTINGS
BP6	BKK92_SUMS
BP7	BCA92 F24Index
BP7	BKK92_POSTINGS Index
BP7	BKK92_SUMS Index
BP8	BCA_ACCTBAL
BP8	BCA92_RESTART
BP9	BCA_ACCTBAL Index
BP9	BCA92_RESTART Index
BP10	BCA_COUNTER
BP11	BCA_COUNTER Index
BP20	BCA_CN_EVENT
BP21	BCA_CN_EVENT Indexes
BP22	BCA_BANO_DUE
BP22	BCA96
BP23	BCA_BANO_DUE Index
BP23	BCA96 Index
BP24	BCA_CN_PER_ACBAL
BP25	BCA_CN_PER_ACBAL Index
BP28	BCA_TRANSFIG
BP29	BCA_TRANSFIG~ Index
BP37	BCA_CNSP_ACCT Indexes
BP41	BCA_GL_PAYMITEM Index
BP48	BCA_CONTRACT
BP49	BCA_CONTRACT Index
BP8K0	BCA_PAYMITEM_ENQ
BP8K0	BCA_PAYMITEM
BP8K0	BCA_PO_IT
BP8K1	BCA_PAYMITEM Index
BP8K3	BCA_PAYMITEM Index 3
BP8K5	BCA_PAYMITEM Index 1
BP8K7	BCA_PAYMITEM Index 2
BP8K9	BCA_PAYMITEM Index 4
BP32K1	BCA_CN_LINK Indexes
BP32K3	BCA_CN_LINK Indexes
BP32K5	BCA_CN_LINK

Table 8: Table/Index Buffer Pool Assignment