



Seewah Chan  
Leticia Cruz  
Veng Ly  
Howard Poole  
Christopher Williams

# IBM Systems for SAP Business Intelligence: 25 Terabyte Scalability Study

Jupiter was a joint project between IBM® and SAP®<sup>1</sup>. The purpose of the benchmark project was to scale a database from 5 to 25 terabytes with little or no decrease in performance. The products in the project focused on SAP NetWeaver® Business Intelligence and SAP NetWeaver BI Accelerator<sup>2</sup>, implemented on an IBM server infrastructure. Our solution demonstrates the capabilities of the accelerator in a very large IBM configuration. We tested scalability with increasing data volumes and resources.

This Redpaper describes the implementation and results of our project, including:

- ▶ The Jupiter solution configuration
- ▶ The SAP NetWeaver BI database load in three data sizes: 5 TB, 15 TB, and 25 TB
- ▶ Test case scenarios for SAP NetWeaver BI Accelerator index creation and multi-user query loads, using the three data sizes with different accelerator configurations
- ▶ The test results, which were validated by an independent third party, WinterCorp<sup>3</sup>
- ▶ A description of the setup and tuning techniques that we used for our large SAP Net Weaver BI Accelerator solution

This benchmark was performed at the Advanced Client Technology (ACT!) System z™ Benchmark Center located in Poughkeepsie, New York. The duration of the project was six months.

<sup>1</sup> © SAP AG 2008. All rights reserved.”

<sup>2</sup> The SAP NetWeaver BI Accelerator is an application that runs on Blade servers with Linux® SuSE on an Intel® Xeon using 64-bit architecture. It requires an SAP BI Accelerator License, SuSE Linux, and a shared file system, such as IBM General Parallel File System™ (GPFS™).

<sup>3</sup> <http://www.wintercorp.com/>

## Background

To be more competitive in a dynamic business environment, large global companies are focused on improving their ability to identify and react to business changes. This responsiveness demands that operational personnel are able to quickly identify any developing business issues. These include detecting an inventory problem, identifying fraudulent purchases in real time, or identifying preferred customers at every point of contact. All of these require that personnel across the lines of business have the correct information, at the correct time, to make the right business decisions. To satisfy this need for information, business intelligence (BI) solutions are evolving to become more proactive, delivering actionable information to the front lines of business operations.

To meet the challenges of an operational BI environment, System z offers many advantages to ensure information availability for all. The System z environment is able to manage workloads from thousands of users, while securing data from external and internal threats, a critical capability for every company. System z environments offer a highly reliable, available, and manageable database infrastructure to support huge query workloads that are typical of an operational BI environment.

Leveraging a mainframe's self-management characteristics, system workload control, and overall cost efficiencies can significantly improve the business value of a solution. With its unique scalability and security, combined with its data availability characteristics, System z provides the capability to seamlessly create an operational BI environment that meets the business requirements of today's 24/7 economy.

Combining these strengths with the performance enhancements of the IBM Systems Solution for SAP NetWeaver BI Accelerator, companies have the ability to leverage the strengths of the mainframe with the performance and flexibility benefits of a turnkey solution, meeting the demanding requirements of today's business climate.

## Partner profile

SAP AG, an IBM Business Partner, is the world's largest business software company and the third-largest software supplier overall. Founded in 1972 as Systems, Applications, and Products in Data Processing, SAP has a rich history of innovation and growth that has made the company the recognized leader in providing collaborative business solutions for all types of industries in every major market.

SAP has leveraged extensive experiences to deliver a comprehensive range of solutions to empower every aspect of business operations. Organizations of all sizes can use SAP solutions including small business solutions and solutions for midsize companies to reduce costs, improve performance, and gain the agility to respond to changing business needs. SAP has delivered a business platform that unlocks valuable information resources, improves supply chain efficiencies, and builds strong customer relationships.

## Project overview

In support of the operational BI requirements of a business, IBM and SAP worked together to demonstrate the extreme scalability possible with this fully integrated solution. This project incorporated System x™ Blades that provide fast parallel processing of queries, complemented by System z and DB2® for z/OS® as the secure, high availability data repository for the volumes of customer's InfoCube data. System p™ was used to support the SAP BI application software because of its processing capabilities at an economical cost.

This powerful combination of technologies provides an integrated environment capable of handling business intelligence today, and far into the future. This server infrastructure, coupled with IBM System Storage™ DS8300 Turbo disk systems, proved to be a very resilient and highly available configuration over the course of the project. No hardware failures occurred during the six months of this project.

During this project, large volumes of customer and benchmark InfoCube data were loaded into DB2. The InfoCube data in DB2 was then indexed and stored in the BI Accelerator for fast query processing. All queries were directed to the BI Accelerator because the focus of the project was to test the scalability of the BI Accelerator. In this project, no aggregates were used, and all cubes were off-loaded to the BI Accelerator.

The scale of our test environment was several magnitudes larger than the largest BI Accelerator blade configuration size (28 blades) previously tested. We chose to test three database sizes: 5 terabytes (TB), 15 TB, and 25 TB. Our project demonstrated that the SAP NetWeaver BI Accelerator on IBM blades can deliver very fast execution for a wide variety of complex queries with consistent response times. This was achieved by using a highly compressed BI Accelerator index data structure and efficient parallel in-memory query processing.

## Project scope

The project scope for this benchmark focused on these activities:

- ▶ Single-user queries
- ▶ Multi-user queries

The query tests aimed for achieving the same response time as database size and number of users increased. This was implemented using these strategies:

- ▶ Concurrency of query streams to support thousands of users
- ▶ Parallel query activities across multiple IBM blades

Our results demonstrated the ability of SAP NetWeaver BI Accelerator to address the growing requirements of users for ad hoc data analysis on a large scale using an IBM infrastructure.

## Jupiter environment infrastructure

This section describes the Jupiter environment infrastructure that we built at the IBM System z Benchmark Center at Poughkeepsie, New York. It includes our:

- ▶ Hardware and software configuration
- ▶ Network infrastructure

By design, some hardware components in this environment were oversized because we did not have all the information required for capacity planning at the beginning of the project. This large configuration was designed to ensure that we would have no resource constraints during any of the scalability tests.

## Hardware and software configuration

The test system configuration we used to run our test case scenarios had the following four major components, as Figure 1 shows: database server, BI server, BI Accelerator, and query load driver.

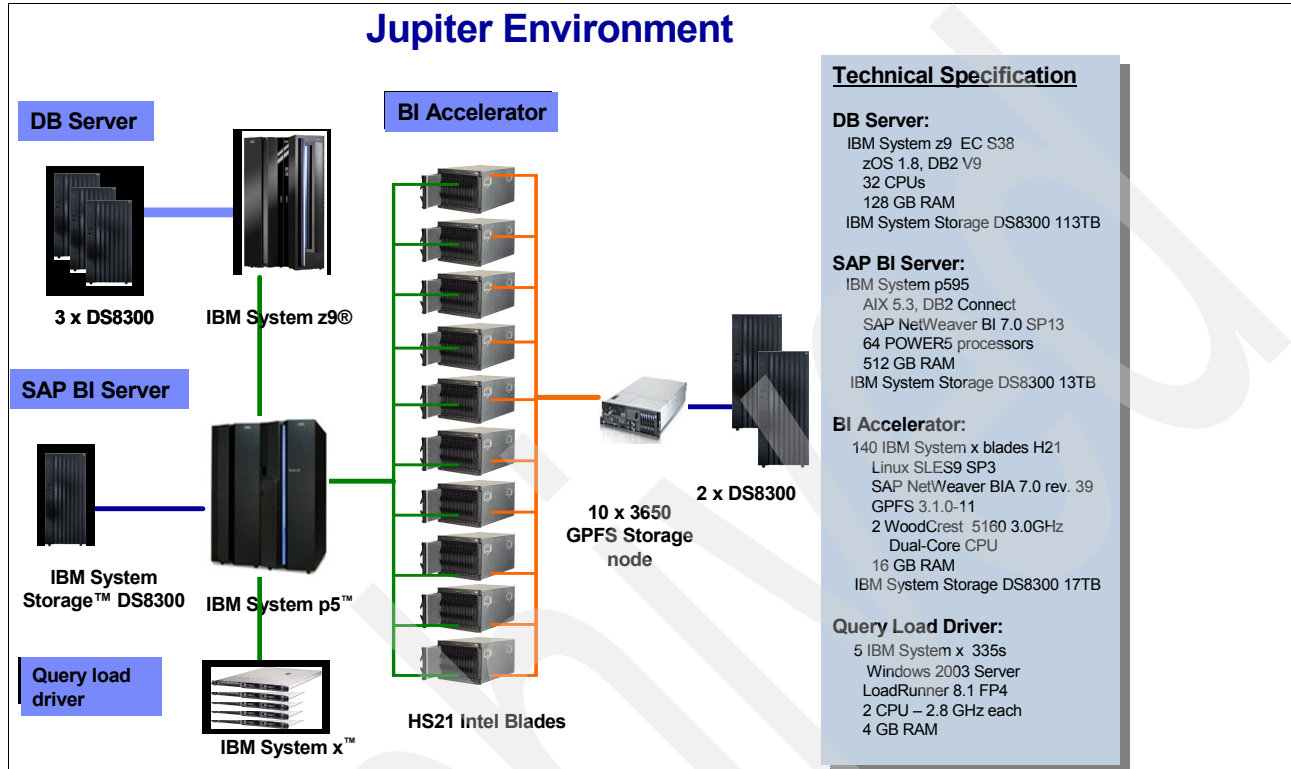


Figure 1 Jupiter environment infrastructure

### Hardware for the server infrastructure

Our hardware consisted of the following systems:

- ▶ The database server was an IBM System z9® Enterprise Class S38 with 32 CPs and 128 GB of central storage.
- ▶ The BI server ran on an IBM System p5™ 595 that had 64 POWER5™ processors and 512 GB RAM.
- ▶ The BI Accelerator was installed on 140 IBM HS21 blades. Each blade had two WoodCrest 5160 3.00 GHz dual-core CPUs and 16 GB RAM, for a total of about 2.2 TB RAM across all blades.

**Note:** One backup blade was configured for every 27 blades. So our configuration consisted of 135 production and five backup blade servers.

- ▶ Five IBM System x 335 workstations with two CPUs, 2.8 GHZ, and 4 GB of RAM were used for the query load driver.
- ▶ Six IBM System Storage DS8300 systems were used in this configuration. Three storage systems were attached to the System z9, one to the System p5, and two to the blades. The RAID-5 configured storage for System z was 129 TB. The RAID-5 configured storage for System p was 14 TB. The RAID-10 configured storage for the blades was 29 TB. That

was the total storage before logical volumes were formatted. The capacities for these storage systems after being initialized were 113 TB, 13 TB, and 17 TB, respectively.

## Software for the hardware components

We installed the following software on each hardware component:

- ▶ IBM System z9: z/OS V1.8, DB2 for z/OS V9
- ▶ IBM System p5: AIX® 5.3, SAP Netweaver BI 7.0 SP13, DB2 Connect™ V9
- ▶ IBM blade servers: Linux SLES9 SP3, SAP Netweaver BI Accelerator 7.0 rev. 39, GPFS 3.1.0-11

## Network configuration

The network configuration for the Jupiter environment used the following technologies, as Figure 2 shows: FICON® Channel, Gigabit Ethernet (GbE), InfiniBand®, and Fibre Channel.

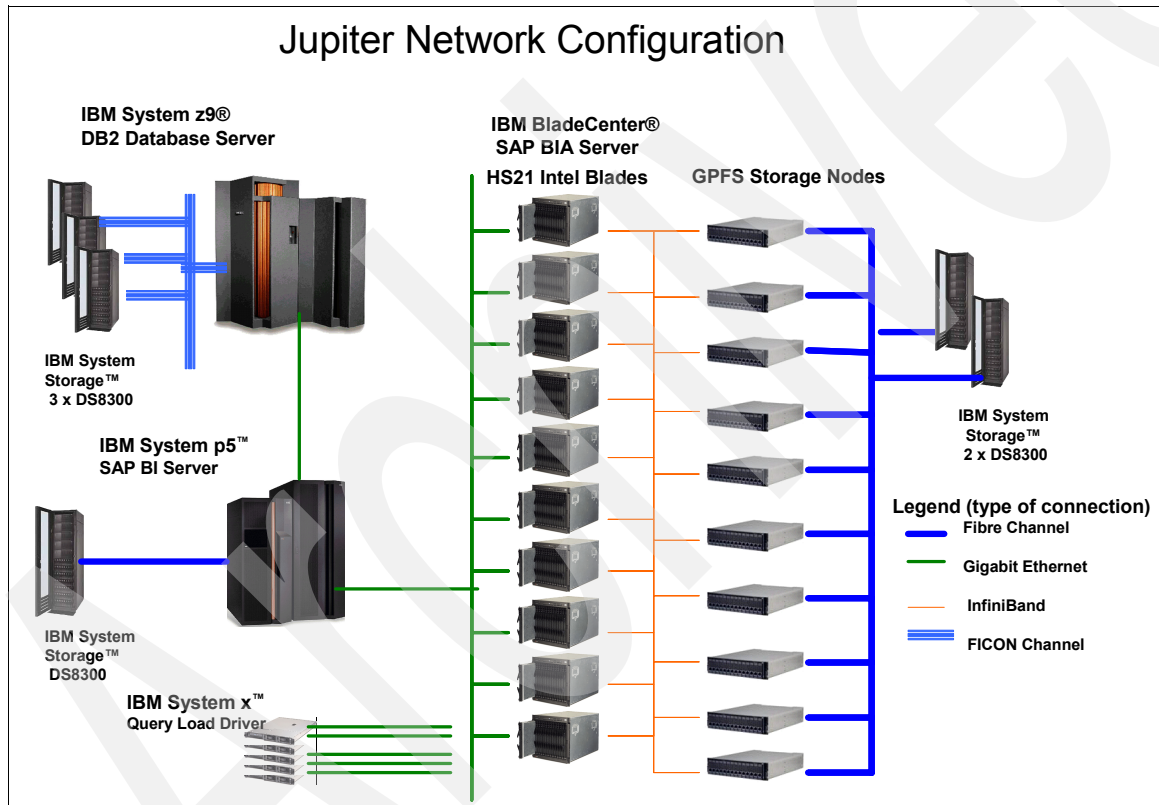


Figure 2 Jupiter network configuration

### Connection between System z and DS8300

Twenty-four FICON links connected System z and the three IBM System Storage DS8300 Turbo disk storage systems (eight connections between System z and each DS8300 storage system). They were critical for writing data to the DB2 database during InfoCube load and reading data during the BI Accelerator index creation.

### Connection between System p and DS8300

Eight Fibre Channel lines connected System p and the DS8300 storage system. Multipath Subsystem Device Driver (SDD) was used. This enabled parallel access to the file systems

on the DS8300 storage system. Because the input files for the InfoCube load resided on this storage system, getting the maximum bandwidth to the storage system was critical.

### **Connection between System p and System z**

Eight 10GbE lines connected System p and System z. Although we could have used fewer than eight connections, the precise data rate requirement was not known at the beginning of the project. It was safer to over-configure than to under-configure, to ensure that there were no network constraints during InfoCube load and during BI Accelerator index creation phases. The data rate was not expected to be close to the aggregated bandwidth of these eight connections. Multiple connections mitigated (but did not totally prevent) delays due to latency from channel busy, hardware implementation, and physical media.

To simplify the environment, our System p was configured as a single system image. Because multiple SAP application servers can run in one partition, our Jupiter project employed multiple instances for a more realistic customer environment. This affected our network implementation, because each SAP application server can use only one SAPDBHOST connection, which is usually associated with one physical network connection. To make use of all eight connections, we combined them into one virtual connection using a virtual IP address (VIPA).

This approach utilizes network trunking, such as EtherChannel or MultiPath. Because EtherChannel is supported on System p but not on System z, we instead used a MultiPath network connection, which has similar functionality to the EtherChannel. Example 1 on page 21 shows our setup definition of static multipath. Each connection was configured with its own subnet on separate VLANs. This ensured that traffic was sent and returned over the same physical connection, and also provided better analysis and monitoring of performance data. Also, a jumbo frame size was configured for the large block sizes transmitted.

For load balancing, an algorithm distributes traffic to all connections when several are combined into one virtual connection. This method provided higher availability because another available physical connection associated with the virtual connection could be used if one of the physical connections was down. For details and guidance on high availability network implementation, refer to *High Availability for SAP on IBM System z Using Autonomic Computing Technologies*, SC33-8206.

### **Connection between System p and blade cluster**

One 10GbE line connected System p to our network switch, and an EtherChannel of four GbE connections from the switch to each blade chassis. There was one GbE interface for each of the 14 blades within the chassis for blade-to-blade connections. Data traffic between System p and the BI Accelerator cluster of blades was mainly through Remote Function Calls (RFCs). A standard Maximum Transmission Unit (MTU) size of 1500 was used because the data package traffic was small.

### **Connection between blade cluster and DS8300**

A General Parallel File System (GPFS) was mounted and accessible to all 140 blades. We expected a high rate of data traffic, so we selected GPFS over a storage area network (SAN) file system for this test. Generally, a SAN is effective with 64 blades or fewer. As the number of blades scales up, GPFS topology with its cluster of network shared disk (NSD) server nodes performs better. In our large environment with 10 BladeCenter® chassis and 140 blades in total, 10 GPFS NSD servers were employed. They were System x3650 machines (two dual-core CPUs at 3.0 GHz with 4 GB RAM) running Linux SLES9 SP3.

Because InfiniBand (IB) has better bandwidth and latency than GbE, we chose IB for the network between the blades and the GPFS NSD servers. Each configured HS21 blade chassis had two embedded Cisco SDR IB switches. Blades 1 through 7 were configured to

use the first IB switch, and blades 8 through 14 were on the second one, to have a non-blocking configuration. Each SDR IB daughter card on each blade was rated at 10 Gbps, or 4x, where  $x = 2.5$  Gbps. Each SDR IB switch had two external 12x ports and two external 4x ports, with an aggregate bandwidth of 80 Gbps. Each GPFS NSD server had 4x or 10 Gbps IB connection.

On the other end, the 10 GPFS NSD servers were connected to GPFS storage (DS8300) via twenty 4 Gbps Fibre Channel connections, two connections per GPFS NSD server.

Because each blade had GbE and IB interfaces, blade-to-blade communication could use GbE or IB. In our case, IB was mainly used for reading and writing to the GPFS file system and GbE was used for blade-to-blade communication.

## SAP NetWeaver BI database and InfoCube load

One of the major challenges for project Jupiter was to build a large, complex (25 TB) BI database within a very short time. According to the plan, a minimum InfoCube data load rate of 1 TB per day was required to meet the aggressive schedules. In the following sections, we describe the database structures and the InfoCube load techniques that we implemented to meet the project requirements.

### SAP NetWeaver BI database

The four most relevant objects used for SAP NetWeaver BI are the master data, persistent staging area (PSA), data store (DSO), and InfoCubes (or simply cubes).

For our Jupiter project, three database sizes were created and populated to supply the data volume requirements for the suite of tests. These were 5 TB, 15 TB, and 25 TB.

In reference to the BI database characteristics, it is important to note that:

- ▶ The growth in database size was reached by increasing the size of the InfoCubes only. The data volume for PSA, DSO, and master data remained constant.
- ▶ The data volume refers to cubes (F fact table and indexes) only. The actual SAP NetWeaver BI database size was larger, if we include the statistics for PSA, DSO, and master data.
- ▶ DB2 data compression was not enabled for this environment. Because the focus of the test was the performance scalability of the BI Accelerator, and because this feature is not available in all database platforms, SAP opted not to enable this DB2 feature.

### BI database InfoCube load

The initial build of the SAP NetWeaver BI database involved the population of master data, PSA, DSO, and InfoCubes. Several challenges were encountered during this phase of the study. Some were caused by the level of DB2 code that we had installed at the time. Others were due to the characteristics of the database, such as the large volume of data being inserted and the concurrency of the jobs during data population.

The execution of this project took six months. As standard benchmark practice, the software code level was not updated from the initial level unless we encountered a defect. However, due to the duration of this project, we did apply a few maintenance updates.

The focus of the study was to showcase the performance scalability of the BI Accelerator in various data volumes. However, the amount of time needed to load the InfoCubes was a significant portion of the project schedule. Thorough planning for disk storage allocation, layout, and backup and restore procedures were essential to meet the demanding milestones.

As part of the pre-benchmark planning activities, we had calculated the total storage required to hold the 25 TB database plus two backup copies. We configured additional disk space for other DB2 tables, z/OS systems, and monitoring data.

We utilized IBM Storage Management System (SMS) to manage and allocate storage space needed for the DB2 data and other system data sets such as log and work files. We defined a dataclass for *extended addressability*. This enabled us to allocate space for linear VSAM data sets larger than 4 GB, which most of the tablespace partitions required.

In the 25 TB BI database, 78 InfoCubes were populated with a total of 30 billion rows, all of which were loaded into the BI Accelerator. The cubes included:

- ▶ Thirty cubes for 10 years of data each for sales orders, billing, and delivery
- ▶ Forty-eight cubes for 48 calendar months for customers

Although the focus of the BI Accelerator tests was on the 78 InfoCubes, more than 32,300 DB2 tables were built on DB2 for z/OS. The 78 cubes comprised 80–90% of the total database size. The largest table group was about 425 GB in size, with nearly 500 million rows per table. This table also had the longest average row length.

During the InfoCube load process, we overcame some of the challenges by implementing changes to the DB2 objects and by applying maintenance to the DB2 code. The modifications to the definition of InfoCube and PSA tables and indexes were as follows:

- ▶ Increasing the DSSIZE from 4 GB to 8 GB
- ▶ Reallocating tables and indexes to 32 K bufferpools
- ▶ Increasing primary allocation of DB2 tablespaces to ensure availability of space during InfoCube load and to minimize the overhead associated with VSAM secondary extents

These enabled us to successfully build the 5 TB, 15 TB, and 25 TB BI databases. The 5 TB and 15 TB databases were similar to the 25 TB BI database in structure, except for scaled-down InfoCube sizes.

For the SAP and DB2 BI database objects, various bufferpools with page sizes of 4 K, 8 K, 16 K, and 32 K were used. For example, DB2 bufferpool BP0 was used for DB2 system objects, BP1 for 4 K sortworks, BP2 for tables, and BP3 for indexes. The bufferpools with 8 K and 16 K pages were used for other SAP system objects.

On the other hand, the bufferpools with 32 K pages were allocated for InfoCubes, PSA tables and indexes, and 32 K DB2 sortworks. The use of 32 K page tablespaces minimizes the amount of leftover space per page for rows with an average length of a few hundred to a thousand bytes. This could also reduce the amount of getpage activity for selects during the InfoCube and BI Accelerator index loads. For example, if there are 910 bytes per row on a table, then with a 32 KB page size, 36 rows can fit on a page. On the other hand, if a 4 KB page is used for this table, the same number of rows would reside on nine pages, because each page can hold only four rows.

In addition, we took advantage of the DB2 bufferpool page fix feature by setting PGFIX to YES. For I/O intensive workloads, this could reduce page fixing overhead. However, the trade-off was higher demand on real memory usage. Because our z9 processor had plenty of real storage, the benefits of PGFIX=YES outweighed the costs.



To see a summary of the bufferpool setup, refer to the following tables in “Appendix: BI Accelerator scalability details” on page 21:

- ▶ Table 5 on page 26
- ▶ Table 6 on page 26
- ▶ Table 7 on page 27
- ▶ Table 8 on page 27

InfoCube loads have characteristics similar to an insert-intensive batch workload. Two common types of overhead in DB2 for this type of workload are locking and logging. To minimize these, we implemented the following DB2 database tuning tips:

- ▶ All secondary indexes for the F fact tables were dropped prior to the InfoCube loads. The secondary indexes were rebuilt, using CREATE INDEX SQL, after the F fact tables were populated.
- ▶ PSA (staging) tables were defined to use PAGE instead of ROW locks to minimize the number of locks acquired.
- ▶ MEMLIMIT for the IRLM PROC, Lock Manager, was raised to 16 GB to accommodate the large number of locks held.
- ▶ PSA table entries were deleted using the LOAD utility with REPLACE DD DUMMY options for fast PSA cleanup.

Other tuning options might exist to improve the InfoCube load rate, but we did not pursue them after we successfully exceeded the minimum load rate required for the project.

## Test case scenarios

The goal of the Jupiter project was to prove the scalability of a BI Accelerator for increasing data volumes and increasing blade resources. The test plan consisted of a suite of tests on:

- ▶ Three database sizes (5 TB, 15 TB, and 25 TB)
- ▶ Various blade configurations (27, 81, and 135 blades)
- ▶ The BI Accelerator index creation
- ▶ A multi-user query workload

Note that for every 27 blades, one backup blade was configured. This brought our total blade requirements to 28, 84, and 140 for the three database sizes.

## SAP NetWeaver BI Accelerator index creation

After all the data was populated into the InfoCubes in a DB2 database on System z, the next step was the BI Accelerator index creation. This phase read data from a row-based database and transformed it into in-memory column-based files in a cluster of blades. In the Jupiter study, all InfoCube data from a DB2 database was indexed to the accelerator blades before it was available for query processing.

## Objectives

The purpose of the accelerator index creation test scenarios was to measure key performance indicators (KPIs) on 5 TB, 15 TB, and 25 TB database size points with blade resources of 27, 81, and 135 systems, respectively. These KPIs were:

- ▶ Elapsed time to create indexes into the SAP NetWeaver BI Accelerator
- ▶ Throughput rate for accelerator index creation
- ▶ CPU utilization on BI server and blade servers

## Description

Because the goal of BI Accelerator index creation was to read data from a database server to a GPFS on blades, certain sizing and configuration settings had to be considered. Proper BI Accelerator blade memory sizing was an important consideration. SAP note 917803 provides an official sizing estimate and guideline for blades. In this study, 135 blades plus five backups (10 HS21 chassis of 14 Woodcrest blades each) were used. Each blade had 16 GB RAM and two dual-core processors at 3 GHz. Thus, 140 blades had about 2.2 TB of memory (RAM).

**Note:** A rule of thumb is to allocate 50% of blade memory for storing the indexed cube data and 50% for the Linux system for processing runtime objects and storing query result sets. This avoids performance impacts from memory swapping to disk.

Table 1 lists our Jupiter BI Accelerator resource requirements.

Table 1 BI Accelerator requirements

Total DB size	RAM	# blades @16 GB RAM	Number of rows
5 TB	432 GB	27 + 1 backup	6 B
15 TB	1296 GB	81 + 3 backups	18 B
25 TB	2160 GB	135 + 5 backups	30 B

In addition, the minimum GPFS size requirement is three times the total blade memory size, which is approximately 6 TB. IBM DS4000™ servers are normally employed when using a smaller number of blades. In this study, the DS8300, an enterprise class storage system, was used.

Our next consideration was the configuration settings for SAP parameters. They can influence data transfer performance: BATCHPARA, PKSIZE, NUMPROC, and SUBPKSIZE. They are defined in the RSDDTREXADMIN table, which contains some of the BI Accelerator administration settings. For details and guidance, refer to the SAP document *Technical Operations Manual for BI Accelerator 7.0*. The following list briefly describes the parameters:

- ▶ BATCHPARA defines the number of parallel background processes used to read data from the fact table of each InfoCube in the database into the data buffer at the SAP NetWeaver BI application server. This parameter essentially governs the degree of parallelism and can require significant system resources.
- ▶ PKSIZE defines the data size in bytes that is transferred from the fact table in the database to the data buffers at the SAP NetWeaver BI application server for each batch process. Parameters PKSIZE and BATCHPARA can result in significant BI system resource requirements. Make sure that a minimum memory size equal to (PKSIZE x BATCHPARA x number of cubes to be indexed in parallel) is available in the BI system.
- ▶ NUMPROC defines the number of dialog processes for each batch process to transfer data from the SAP BI data buffer to the accelerator via Asynchronous Remote Function

Call (aRFC). Ensuring that enough dialog processes and RFC connections are defined is important.

- ▶ SUBPKGSIZE defines the data size, in number of records, to be sent from the SAP BI data buffer to the BI Accelerator through aRFC.

Figure 3 explains the data flow from database server to BI Accelerator relationship of these parameters and the calculation for the required number of Remote Function Call (RFC) connections.

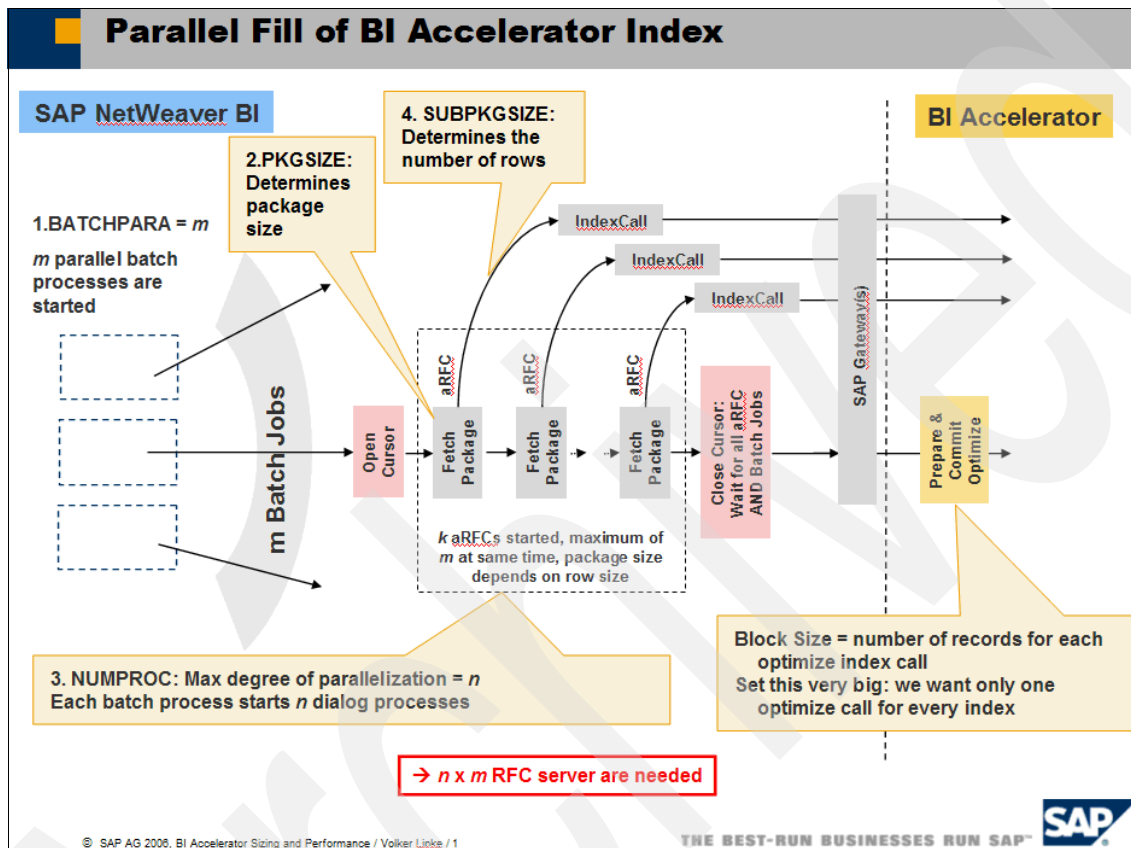


Figure 3 Parallel fill of BI accelerator index

To describe how the parameters influence the BI Accelerator creation process, we use one Jupiter example, an index creation of 10 cubes in parallel with the following settings:

- ▶ BATCHPARA=6
- ▶ PKGSIZE=100,000,000
- ▶ NUMPROC=5
- ▶ SUBPKGSIZE=20,000 rows

During the initial index creation phase, data was read from the F fact tables in the database to data buffers in the SAP BI application server. Because BATCHPARA=6, six concurrent background processes were started for each cube. Each data transfer process moved data in 100 MB chunks, as specified in the PKGSIZE parameter. Because we chose to create indexes for 10 cubes in parallel, a total of 60 background processes ran concurrently, if enough SAP batch work processes were available. For our project, we defined 256 SAP batch work processes.

After the data was in the data buffer of the SAP BI application server, the data package size PKGSIZE was subdivided into SUBPKGSIZE. Each package of 20,000 rows was transferred to

the BI Accelerator through aRFCs employing five dialog processes (NUMPROC). Thus, for each batch process, five dialog processes were defined. Therefore, the total number of actively used dialog processes could have been up to 300 (BATCHPARA x NUMPROC x number of cubes).

In addition, each cube, especially large ones, can be configured to distribute to a set number of blades for indexing, depending on the configured number of parts. The cube is considered to be very large if it contains several hundred million rows. Because different cubes could have different numbers of columns, the number of cells is a more accurate method of deciding whether a cube is large. By default, 100 million cells (number of rows x column key figures) is a threshold to determine that the cube is large, and thus a candidate for horizontal partitioning. Figure 4 illustrates the horizontal partitioning concept.

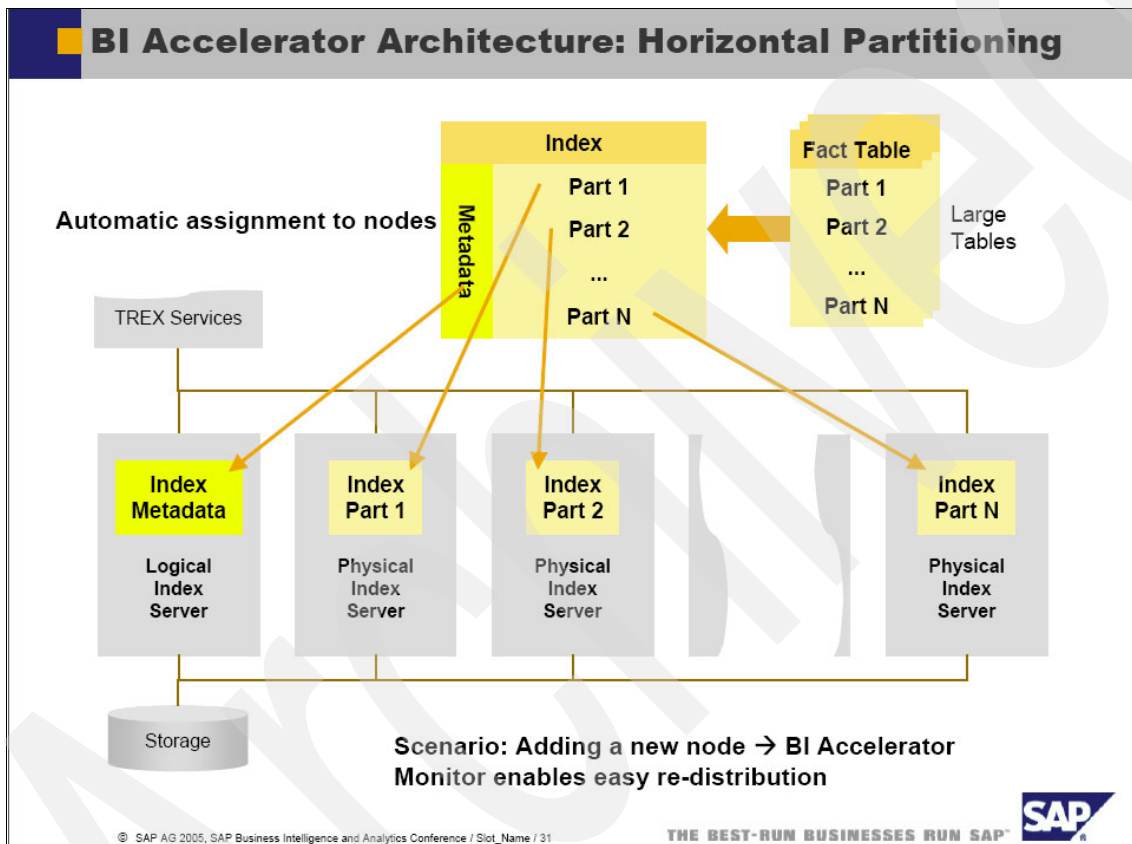


Figure 4 Horizontal partitioning

All InfoCubes used in this study were in this large cube category and were candidates for partitioning. Our study threshold setting was 250 million cells. Certain tests were conducted with 27 parts and others with 40 parts. The number of parts was a trade-off between having more parallel tasks and having more interblade communication traffic. This setting can be implemented with the TREXAdmin tool:

```
Tree > index > scrolling down to the bottom
(max_size=250,000,000, and max_split_parts=40).
```

When the data arrives at the blade, the RFC compressed data is uncompressed. Then, gzip compression is done before writing temporary objects to GPFS, in the form of \*.tmpx files. Each blade had its own unique storage location subdirectories for the same cube. The next phase is to read these temporary objects that were just written to GPFS. Initial indexing and optimizing take place. This uncompresses the data, reorganizes it, and compresses again

with a different algorithm. Lastly, the data is written as a whole to the GPFS as final persistent objects in the form of \*.bin files.

Several examples in “Appendix: BI Accelerator scalability details” illustrate SAP NetWeaver BI Accelerator horizontal partitioning, vertical decomposition, and compression. The examples are 6 through 8 and begin on page 31.

They display a list of BI Accelerator indexed cubes, a cube example with 40 part horizontal partitioning, a list of all columns of an indexed cube, and their corresponding temporary and persistent objects.

The BI Accelerator index creation process was very I/O intensive to the GPFS storage shared by the blades. During the test, we started with the default value of 256 K for the GPFS I/O block size. We added a second DS8300 storage system to handle the high GPFS I/O activity. Our further investigation found that the high GPFS I/O activity was actually due to the large block size of 256 K, so we experimented and determined that blocksize=64K and maxblocksize=64K yielded better GPFS performance.

The two DS8300 storage systems were configured using RAID-10 because of the write intensity of the BI Accelerator index creation. In general, RAID-10 is recommended for write-intensive workloads. However, it requires more disk space.

### Measurement results

Table 2 provides the resulting GPFS data sizes after BI Accelerator index creation for our 5 TB, 15 TB, and 25 TB databases. Note that although the final space used in GPFS is much smaller than configured, more space is actually used for storing BI Accelerator temporary objects. Additionally, more than three times the space is required if backups or other activities are planned. Most importantly, this study shows that from 0.6 TB to 1.3 TB of fact table data per hour can be indexed to the BI Accelerator, which is quite impressive.

*Table 2 Index creation results: throughput, duration, and space*

Database volume size and blade resources	Throughput (terabytes/hr)	Duration (hh:mm:ss)	GPFS size
5 TB index creation 27 blades	0.6	6:40:30	300 GB
15 TB index creation 81 blades	1.1	11:07:07	800 GB
25 TB index creation 135 blades	1.3	15:04:29	1.1 TB

The CPU usage for the database server on System z was rather light during BI Accelerator index creation. The average CPU rates at steady state were 11.84%, 25.00%, and 26.72% for 5 TB, 15 TB, and 25 TB, respectively.

The next three figures show the CPU utilization of SAP BI on System p and SAP NetWeaver BI Accelerator on blades for the three accelerator index creation measurements taken at steady state. The average high load utilization on the charts refers to the average CPU utilization at steady state.

Figure 5 shows that the average CPU utilization during the 5 TB accelerator index creation was 28.72% on System p and 75.32% on blades.

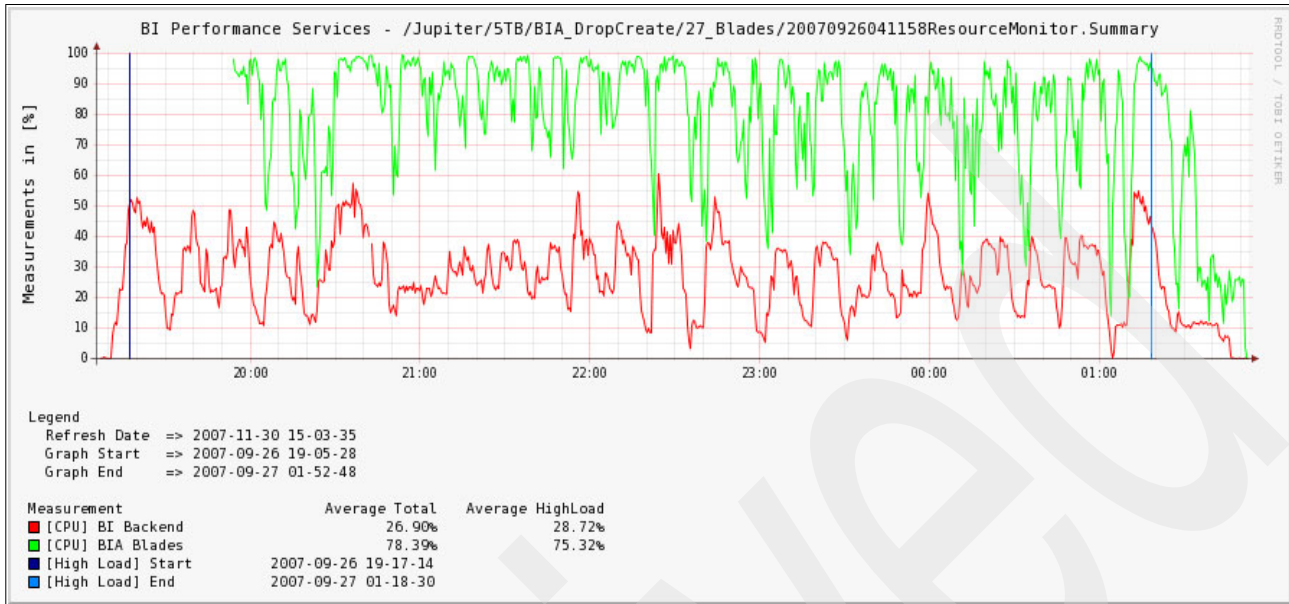


Figure 5 Index creation: average CPU load of 5 TB, 27 blades, 5 InfoCubes parallel

Figure 6 shows the average CPU utilization during the 15 TB BI Accelerator Index creation. These are 61.65% for System p and 69.86% for the blades.

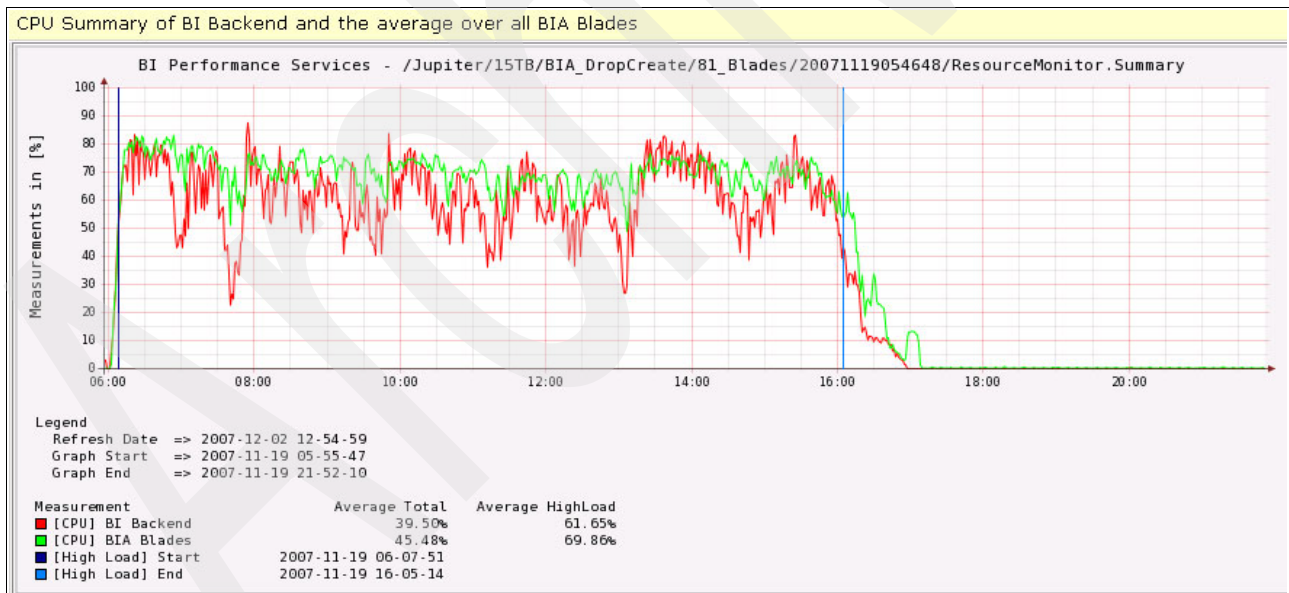


Figure 6 Index creation: average CPU load of 15 TB, 81 blades, 10 InfoCubes parallel



Figure 7 shows the average CPU utilization for the 25 TB BI Accelerator Index creation. These are 75.24% and 57.17% for System p and blades, respectively.

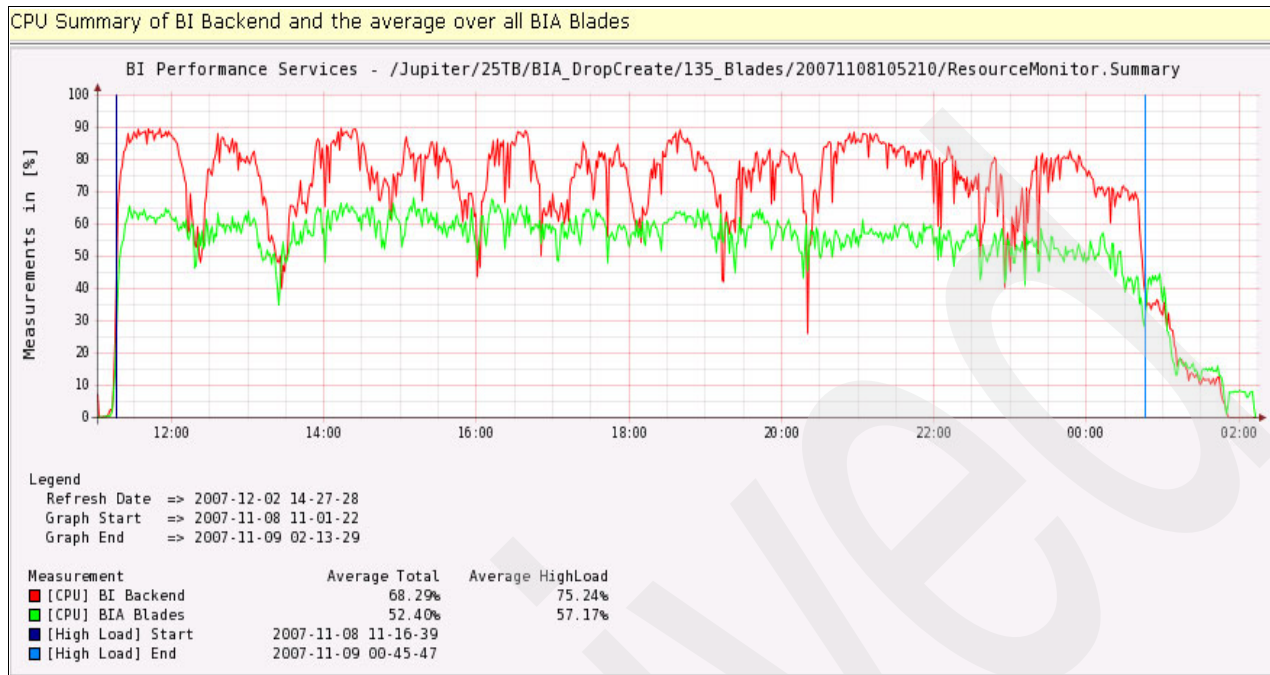


Figure 7 Index creation: average CPU load of 25 TB, 135 blades, 10 InfoCubes parallel

Note that the overall blade utilization figure decreased for each larger database size. This is because we added more blades to the accelerator each time that we enlarged the database.

## Multi-user query

To test multi-user query scenarios, a load driver was used to simulate multiple users issuing queries to the SAP BI system.

### Objectives

The goal for the multi-user query scenarios was to measure the following KPIs for 5 TB on 27 blades, 15 TB on 81 blades, and 25 TB on 135 blades:

- ▶ Query throughput rate in terms of business reports per hour and InfoCube queries per hour
- ▶ Average number of records touched per query in the BI Accelerator
- ▶ Average report response time
- ▶ CPU consumption on blades and BI server

### Description

When a query was issued by a user, SAP BI on System p checked whether the data to be queried had been BI Accelerator indexed. If it had, query processing was directed to the SAP NetWeaver BI Accelerator. Then, if the data was not in BI Accelerator blade memory, it had to be read from GPFS into blade memory. In preparation for query processing, all cube data can be pre-loaded from GPFS into blade memory during index creation or off-shift hours. This increases the speed of query processing by eliminating I/O during production.

For our query measurements, all BI Accelerator indexes were pre-loaded into memory.

Of all the records in the InfoCubes, only selected rows (QDBSEL) are relevant to the query. As with RDBMS, filtering of all records in the InfoCube is done to identify those relevant to the query for further processing. Typically, indexes are used to do this. In this case, BI Accelerator column-based processing and compression have a clear advantage over an RDBMS in identifying and processing the relevant records. The selected records are summarized or aggregated concurrently by a number of blades, then the records are transferred to an OLAP engine in an SAP BI appserver residing on System p. The number of records transferred is referred to as QDBTRANS. At the SAP BI appServer, server rendering is performed before sending the query report to users.

Note that the queries that we designed for this study had high selectivity, with an average QDBSEL of 26–46 million records per query, for our 25 TB tests. Still, the BI Accelerator does tremendously well with an average response time of less than 20 seconds.

### Measurement results

One benefit to using BI Accelerator is that no explicit tuning for the cubes is required to prepare for query processing. So no aggregates have to be defined and no additional secondary indexes are needed. The number of queries per hour shown in Table 3 is the number of business reports (user queries) per hour.

Table 3 BI Accelerator query test results: throughput, selectivity, response time

DB/blade size	Throughput (queries/hr)	Response time (seconds)	QDBSEL (records)	QDBTRANS (records)	Part provider factor
5 TB/27 blades	100,404	4.5	5,906,534	303	1.69
15 TB/81 blades	100,940	4.2	22,186,286	440	1.68
25 TB/135 blades	100,712	4.2	36,767,954	520	1.69

The average CPU utilization on the database server on System z for this workload at steady state was less than 10%. The averages were 6.7%, 7.2%, and 6.2% for 5 TB, 15 TB, and 25 TB, respectively.

The next three figures show the average CPU load utilization of our SAP BI backend on System p and SAP NetWeaver BI Accelerator on the cluster of blades for the three measurement points during multi-user test scenarios. The average high load is the measured CPU utilization for approximately 15 minutes of peak steady state.

Figure 8 shows that the average CPU high load of the SAP BI backend on System p was 79.29%, and the accelerator on the cluster of blades was 71.69% for 5 TB and 27 blades.



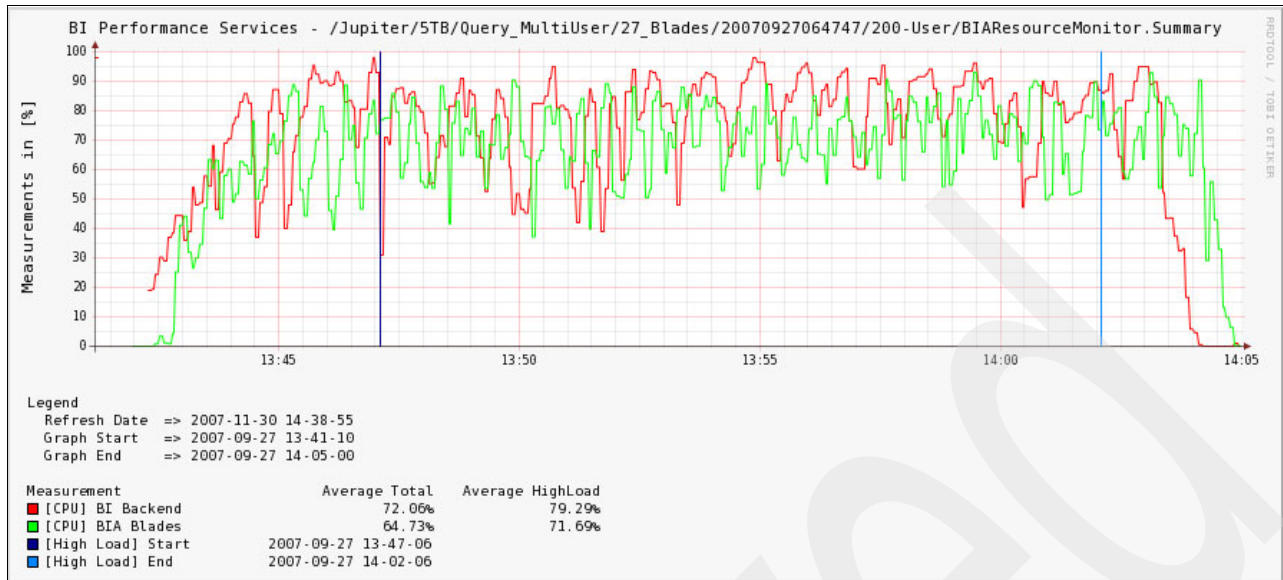


Figure 8 Multiuser reporting: average CPU load 5 TB, 27 blades

Figure 9 shows that the average CPU high load of the SAP BI backend on System p was 87.07% and the accelerator on cluster of blades was 66.85% for 15 TB and 81 blades.

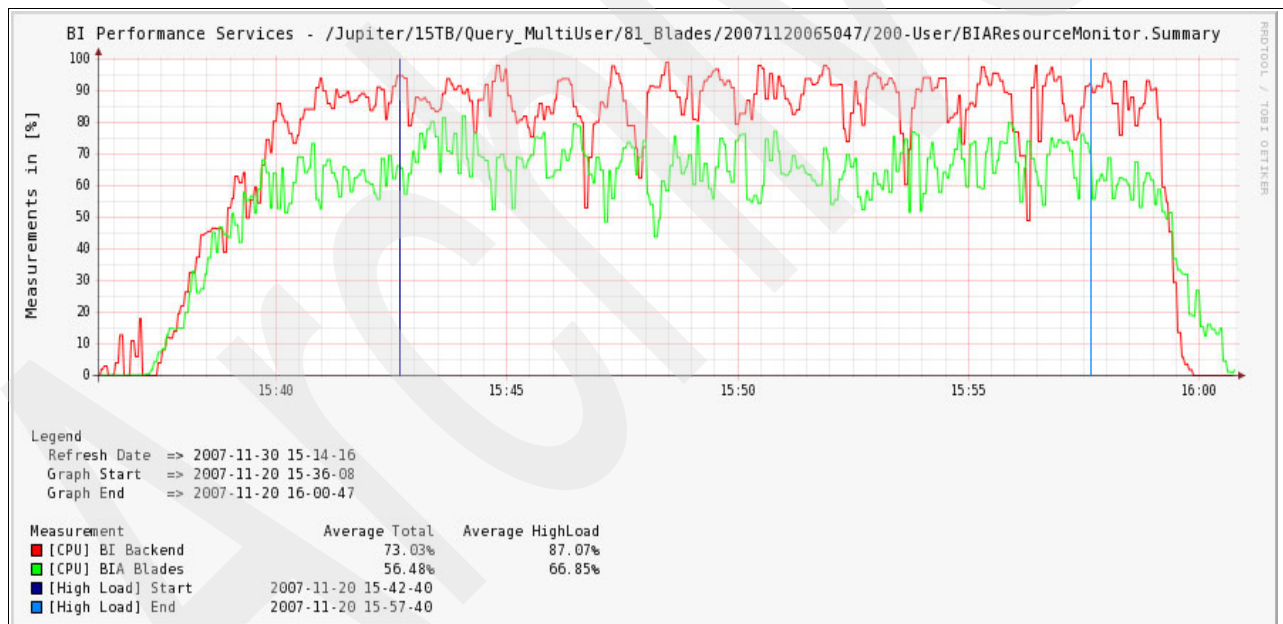


Figure 9 Multiuser reporting: average CPU load 15 TB, 81blades

Figure 10 shows that the average CPU high load of SAP BI backend on System p was 87.99% and the accelerator on the cluster of blades was 60.08% for 25 TB on 135 blades.

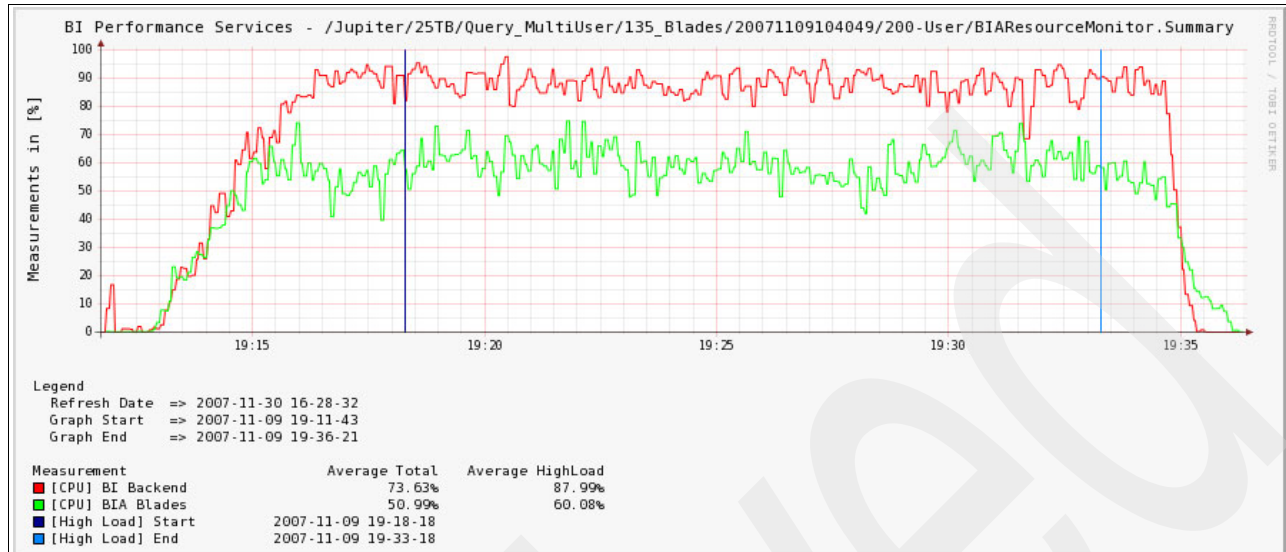


Figure 10 Multiuser reporting: average CPU load 25 TB, 135 blades

## Analysis and conclusions

Project Jupiter demonstrated that DB2 for z/OS coupled with the SAP NetWeaver BI Accelerator is an ideal solution for business information and data warehousing. DB2 for z/OS, as the premiere IBM relational database management system, offers industry-leading performance, scalability, reliability, and manageability.

SAP NetWeaver BI Accelerator with DB2 for z/OS on IBM Server Infrastructure can deliver very fast query execution with consistent response times. This was achieved with the use of highly compressed accelerator index data structures and efficient parallel in-memory query processing. During our project with 100% query off-load to the accelerator, the need for extensive aggregates and database indexes tuning was completely eliminated, while exploiting DB2's strength as the master data repository. Figure 11<sup>4</sup> illustrates the key performance indicators (KPIs) of our project.

<sup>4</sup> This information was taken from a presentation by Thomas Becker of SAP, titled, "JUPITER: 25TB Showcase for SAP NetWeaver BI Accelerator and IBM Server Infrastructure," at the IBM Poughkeepsie Customer Briefing Center on December 13th, 2007.

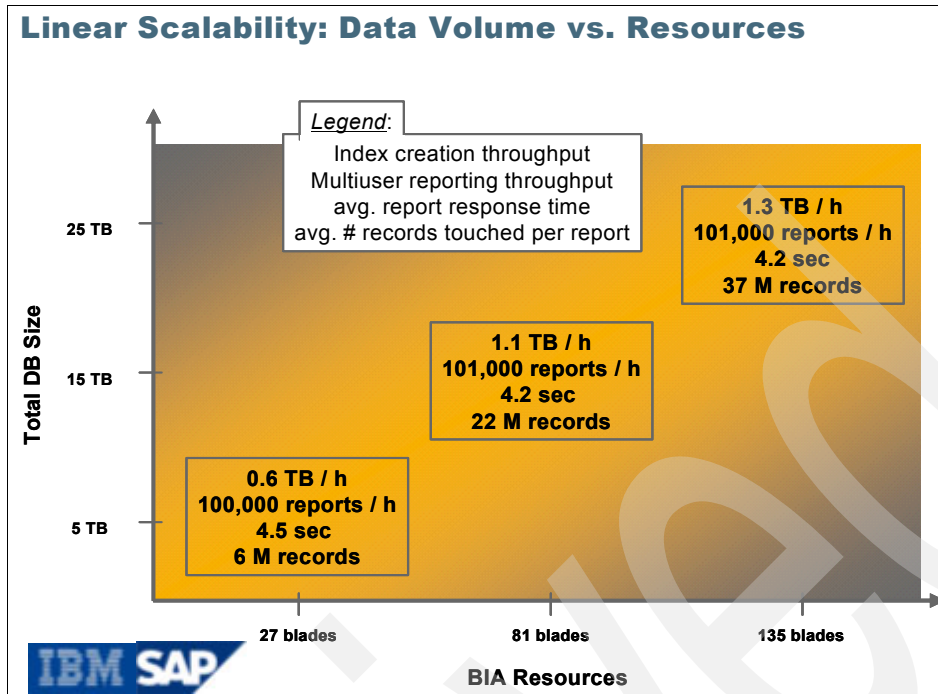


Figure 11 Linear scalability: data volume versus resources

Before our project, the largest supported SAP BI Accelerator configuration was the XLarge size that consisted of up to 28 blades. Our results showed BI Accelerator query scalability up to 135 (plus 5 spare) blades with no observable scalability limits. Likewise, the BI Accelerator Index load rate can be sustained above 1 TB per hour. These impressive throughput results for BI Accelerator index load as well as for query data retrieval will readily meet the stringent requirements of the largest BI customers today.

## Recommendations for configuration

In summary, the primary objective of project Jupiter was to evaluate BI Accelerator scalability. To ensure that no hardware resource constraints were encountered, we took a conservative approach with some of the hardware capacity planning. Because the scale of our test environment was several magnitudes larger than the 28 blades supported prior to our project, there was a high level of uncertainty about the scaling effects. As a result, a few hardware components were highly oversized. Therefore, the configuration used in our project should not be used as a direct guideline for customer environments. See “Help getting started” on page 20, for guidance.

One exception to the highly oversized components was the BI Accelerator sizing used in our project. The general SAP rule-of-thumb for BI Accelerator sizing is to keep blade memory consumption by BI Accelerator indexes below 50% of the total memory on the blade servers. This is to ensure that sufficient memory is available on the blades to allow the query executions without causing performance issues. We adhered to this general guideline for our BI Accelerator blade configuration. The fast consistent query response times achieved during our project have further validated this guideline.

To minimize exposure to known problems, it is important to keep current with SAP support packs and the DB2 maintenance level as identified in SAP notes 364109 (DB2 Put Levels) and 81737 (APAR List).

A time-consuming task is checking that all required program temporary fixes (PTFs) listed in SAP Notes 364109 and 81737 have been applied to a z/OS system. SAP provides a tool called *automated PTF checker* to facilitate the checking. Read SAP note 183311 for details.

## Help getting started

To facilitate the BI Accelerator sizing process, SAP has provided an ABAP™ report that can be used to estimate the memory consumption of the BI Accelerator indexes. Refer to SAP Note 917803 for how to use this tool on the target BI system. Though SAP BI 7.0 is required for the BI Accelerator, the tool can be run on the older SAP BW Releases 3.0, 3.1, and 3.5, as well as BI 7.0. Note that since the tool analyzes all cubes, it may run for a long time and therefore needs to be scheduled accordingly. Contact SAP if there is any issue with the report or information in note 917803.

After a successful run of the tool, you have two options for where to send the report output:

- ▶ E-mail your output to [Sizings@us.ibm.com](mailto:Sizings@us.ibm.com)<sup>5</sup>. The Techline ISV sizing team will determine the appropriate size of blade configuration based on the memory requirement for the BI Accelerator index. An e-mail with the certified e1350 BI Accelerator configurations will be sent to the contact person.
- ▶ Contact an IBM Business Partner. Indicate the client's name and a contact name. The IBM Business Partner will work directly with you to size and deliver the BI Accelerator configurations.

**Note:** For both options, standard SAP BI Accelerator configurations exist in which all options are fixed.

Besides the BI Accelerator blade configuration, it is also important to have a properly sized configuration for the rest of your SAP system landscape, including the database server, application servers, disk storage units, and network topology. For the proper sizing and configuration, contact the IBM technical sales specialist through your local IBM account team.

<sup>5</sup> IBM Corp. 2006. Refer to FAQ: How do I obtain a sizing for an IBM Systems Solution for SAP BI Accelerator (BI Accelerator) from SAP? (<http://www.ibm.com/support/techdocs/atsmastr.nsf/WebIndex/FQ117071>)

## Appendix: BI Accelerator scalability details

This appendix contains the scalability details of Project Jupiter, including definitions and results.

### Network configuration: static multipath

Examples 1 through 3 show our static multipath connection setup between System z and System p. Example 1 shows the TCP/IP profile setup for System z. It defines eight GbE connections and one virtual IP address (VIP) connection.

#### *Example 1 System z TCP/IP profile*

---

```
; Hardware definitions:
;
; CU Ethernet (10GBE) Performance subnet
DEVICE CU041010 MPCIPA
LINK P17E11 IPAQENET CU041010

; CU Ethernet (10GBE) Performance subnet
DEVICE CU051020 MPCIPA
LINK P17E12 IPAQENET CU051020

; CU Ethernet (10GBE) Performance subnet
DEVICE CU061030 MPCIPA
LINK P17E13 IPAQENET CU061030

; CU Ethernet (10GBE) Performance subnet
DEVICE CU071040 MPCIPA
LINK P17E14 IPAQENET CU071040

; CU Ethernet (10GBE) Performance subnet
DEVICE CU081050 MPCIPA
LINK P17E15 IPAQENET CU081050

; CU Ethernet (10GBE) Performance subnet
DEVICE CU091060 MPCIPA
LINK P17E16 IPAQENET CU091060

; CU Ethernet (10GBE) Performance subnet
DEVICE CU0A1070 MPCIPA
LINK P17E17 IPAQENET CU0A1070

; CU Ethernet (10GBE) Performance subnet
DEVICE CU0B1080 MPCIPA
LINK P17E18 IPAQENET CU0B1080

; VIPA interface
DEVICE VDEV1 VIRTUAL 0
LINK VIPA1 VIRTUAL 0 VDEV1

HOME
129.40.178.17 PELP17
129.40.241.11 P17E11
129.40.242.12 P17E12
129.40.243.13 P17E13
129.40.244.14 P17E14
129.40.245.15 P17E15
```

```

129.40.246.16 P17E16
129.40.247.17 P17E17
129.40.248.18 P17E18

129.40.184.1 VIPA1

GATEWAY
;
; Direct Routes - Routes that are directly connected to my interfaces.
;
; Network      First Hop  Link Name  PacketSize Subnet Mask Subnet Value
129.40        =         P17E11    9000        0.0.255.0 0.0.241.0
129.40        =         P17E12    9000        0.0.255.0 0.0.242.0
129.40        =         P17E13    9000        0.0.255.0 0.0.243.0
129.40        =         P17E14    9000        0.0.255.0 0.0.244.0
129.40        =         P17E15    9000        0.0.255.0 0.0.245.0
129.40        =         P17E16    9000        0.0.255.0 0.0.246.0
129.40        =         P17E17    9000        0.0.255.0 0.0.247.0
129.40        =         P17E18    9000        0.0.255.0 0.0.248.0

; Start all the defined devices.

START CU041010
START CU051020
START CU061030
START CU071040
START CU081050
START CU091060
START CU0A1070
START CU0B1080

```

Example 2 shows **netstat -in** information for System p. It also shows eight GbE connections using an MTU size of 9000 for our network interface.

*Example 2 System p network interface information*

```

r60t01:napadm 13> netstat -in
Name Mtu Network Address Ipkts Ierrs Opkts Oerrs Coll
en0 9000 link#2 0.11.25.7b.5.9f 326810878 0 141344687 1 0
en0 9000 129.40.243 129.40.243.3 326810878 0 141344687 1 0
en1 9000 link#3 0.11.25.7b.4.f3 357858123 0 153794074 2 0
en1 9000 129.40.241 129.40.241.1 357858123 0 153794074 2 0
en2 9000 link#4 0.11.25.7b.7.45 293587854 0 126895182 2 0
en2 9000 129.40.242 129.40.242.2 293587854 0 126895182 2 0
en3 9000 link#5 0.11.25.7b.5.71 227762617 0 98547230 2 0
en3 9000 129.40.244 129.40.244.4 227762617 0 98547230 2 0
en7 9000 link#6 0.11.25.7b.5.47 296920097 0 127977123 2 0
en7 9000 129.40.247 129.40.247.7 296920097 0 127977123 2 0
en8 9000 link#7 0.11.25.7b.a.42 325638428 0 140604715 2 0
en8 9000 129.40.245 129.40.245.5 325638428 0 140604715 2 0
en9 9000 link#8 0.11.25.7b.4.d9 379675884 0 162570375 2 0
en9 9000 129.40.248 129.40.248.8 379675884 0 162570375 2 0
en10 9000 link#9 0.11.25.7b.5.5c 407537314 0 173067762 2 0
en10 9000 129.40.246 129.40.246.6 407537314 0 173067762 2 0
en12 1500 link#10 0.14.5e.b7.73.92 10745001 0 8899014 2 0
en12 1500 129.40.28 129.40.28.10 10745001 0 8899014 2 0
en11 1500 link#11 0.11.25.7b.4.d2 980950395 0 172190608 0 0
en11 1500 129.40.49 129.40.49.200 980950395 0 172190608 0 0
lo0 16896 link#1 2509044 0 2512091 0 0

```

```

lo0 16896 127          127.0.0.1          2509044  0 2512091  0  0
lo0 16896 ::1         2509044  0 2512091  0  0

```

Example 3 shows static routing of the GbE connections to a VIPA connection. This is defined through smitty with the multipath option and policy=weighed round-robin (RR) algorithm and weight=1. Thus, network traffic is sprayed to all eight GbE connections evenly for load balancing.

*Example 3 System p route with multipath option*

Destination	Gateway	Flags	Refs	Use	If	Exp	Groups
Route Tree for Protocol Family 2 (Internet):							
default	129.40.49.254	UG	6	64967	en11	-	-
127/8	127.0.0.1	U	26	2499406	lo0	-	-
129.40.28.0	129.40.28.10	UHSb	0	0	en12	-	- =>
129.40.28/27	129.40.28.10	U	2	8898501	en12	-	-
129.40.28.10	127.0.0.1	UGHS	0	7	lo0	-	-
129.40.28.31	129.40.28.10	UHSb	0	4	en12	-	-
129.40.49.0	129.40.49.200	UHSb	0	0	en11	-	- =>
129.40.49/24	129.40.49.200	U	835	172156947	en11	-	-
.....							
129.40.184.1/32	129.40.243.13	UG	57	141346068	en0	-	- =>
129.40.184.1/32	129.40.241.11	UG	57	153795620	en1	-	- =>
129.40.184.1/32	129.40.242.12	UG	58	126899865	en2	-	- =>
129.40.184.1/32	129.40.244.14	UG	58	98547899	en3	-	- =>
129.40.184.1/32	129.40.245.15	UG	58	140606360	en8	-	- =>
129.40.184.1/32	129.40.247.17	UG	58	127978194	en7	-	- =>
129.40.184.1/32	129.40.246.16	UG	57	173077376	en10	-	- =>
129.40.184.1/32	129.40.248.18	UG	58	162571986	en9	-	-
.....							
129.40.248/24	129.40.248.8	U	1	1	en9	-	-
129.40.248.8	127.0.0.1	UGHS	0	7	lo0	-	-
129.40.248.255	129.40.248.8	UHSb	0	4	en9	-	-

## SAP BI database

This section contains information about our InfoCube specifications and our bufferpool parameters. Table 4 shows the row length and total rows for each database size.

Table 4 Database summary of InfoCubes

Table name	Average row length	Total rows 5 TB	Total rows 15.3 TB	Total rows 25 TB
/BIC/FJSDBLC01	590	60,000,000	270,000,000	495,000,000
/BIC/FJSDBLC02	590	60,000,000	275,000,000	500,000,000
/BIC/FJSDBLC03	590	60,000,000	275,000,000	499,600,000
/BIC/FJSDBLC04	590	60,000,000	282,800,000	507,800,000
/BIC/FJSDBLC05	590	60,000,000	272,000,000	497,000,000
/BIC/FJSDBLC06	590	60,000,000	282,200,000	507,000,000
/BIC/FJSDBLC07	590	60,000,000	275,000,000	500,000,000
/BIC/FJSDBLC08	590	60,000,000	283,000,000	508,000,000

Table name	Average row length	Total rows 5 TB	Total rows 15.3 TB	Total rows 25 TB
/BIC/FJSDBLC09	590	60,000,000	270,000,000	493,200,000
/BIC/FJSDBLC10	590	60,000,000	283,000,000	506,800,000
/BIC/FJSDDL01	760	60,000,000	270,000,000	499,000,000
/BIC/FJSDDL02	760	60,000,000	280,000,000	509,000,000
/BIC/FJSDDL03	760	60,000,000	270,086,933	499,200,000
/BIC/FJSDDL04	760	60,000,000	280,000,000	509,000,000
/BIC/FJSDDL05	760	60,000,000	275,000,000	504,000,000
/BIC/FJSDDL06	760	60,000,000	280,013,811	509,000,000
/BIC/FJSDDL07	760	60,000,000	270,000,000	499,200,000
/BIC/FJSDDL08	760	60,000,000	275,000,000	504,000,000
/BIC/FJSDDL09	760	60,000,000	275,000,054	504,200,000
/BIC/FJSDDL10	760	60,000,000	270,000,000	499,000,000
/BIC/FJSDORC01	910	55,000,000	265,000,000	481,200,000
/BIC/FJSDORC02	910	60,000,000	275,000,000	491,400,000
/BIC/FJSDORC03	910	60,000,000	270,000,000	486,400,000
/BIC/FJSDORC04	910	60,000,000	270,000,000	486,400,000
/BIC/FJSDORC05	910	60,000,000	270,000,000	486,200,000
/BIC/FJSDORC06	910	60,000,000	275,000,000	491,400,000
/BIC/FJSDORC07	910	60,000,000	265,000,000	486,400,000
/BIC/FJSDORC08	910	60,000,000	275,000,000	491,400,000
/BIC/FJSDORC09	910	60,000,000	269,999,999	486,400,000
/BIC/FJSDORC10	910	60,000,000	275,000,000	491,400,000
/BIC/FZJB_07_01	653	55,000,000	179,999,998	280,000,000
/BIC/FZJB_07_02	653	60,000,000	185,000,000	285,000,000
/BIC/FZJB_07_03	653	60,000,000	185,000,000	285,000,000
/BIC/FZJB_07_04	653	55,000,000	180,000,000	280,000,000
/BIC/FZJB_07_05	653	60,000,000	185,004,657	285,000,000
/BIC/FZJB_07_06	653	60,000,000	185,000,000	285,000,000
/BIC/FZJB_07_07	653	60,000,000	185,000,000	285,000,000
/BIC/FZJB_07_08	653	60,000,000	185,000,000	285,000,000



Table name	Average row length	Total rows 5 TB	Total rows 15.3 TB	Total rows 25 TB
/BIC/FZJB_07_09	653	60,000,000	185,000,000	285,000,000
/BIC/FZJB_07_10	653	55,000,000	180,000,000	280,000,000
/BIC/FZJB_07_11	653	60,000,000	185,000,000	285,000,000
/BIC/FZJB_07_12	653	60,000,000	185,000,000	285,000,000
/BIC/FZJB_08_01	653	55,000,000	180,000,000	278,400,000
/BIC/FZJB_08_02	653	60,000,000	185,000,000	283,400,000
/BIC/FZJB_08_03	653	60,000,000	185,000,000	283,800,000
/BIC/FZJB_08_04	653	55,000,000	180,000,000	278,400,000
/BIC/FZJB_08_05	653	60,000,000	187,553,848	284,200,000
/BIC/FZJB_08_06	653	60,000,000	185,000,000	283,800,000
/BIC/FZJB_08_07	653	60,000,000	185,000,000	283,200,000
/BIC/FZJB_08_08	653	60,000,000	185,000,000	283,400,000
/BIC/FZJB_08_09	653	60,000,000	185,000,000	283,600,000
/BIC/FZJB_08_10	653	55,000,000	175,000,000	274,000,000
/BIC/FZJB_08_11	653	60,000,000	185,000,000	283,600,000
/BIC/FZJB_08_12	653	60,000,000	185,000,000	283,600,000
/BIC/FZJB_09_01	653	55,000,000	180,000,000	280,000,000
/BIC/FZJB_09_02	653	60,000,000	185,000,000	280,000,000
/BIC/FZJB_09_03	653	60,000,000	185,000,000	285,000,000
/BIC/FZJB_09_04	653	55,000,000	180,000,000	280,000,000
/BIC/FZJB_09_05	653	60,000,000	185,000,000	285,000,000
/BIC/FZJB_09_06	653	60,000,000	185,000,000	285,000,000
/BIC/FZJB_09_07	653	60,000,000	185,000,000	285,000,000
/BIC/FZJB_09_08	653	60,000,000	185,000,000	285,000,000
/BIC/FZJB_09_09	653	60,000,000	185,000,000	285,000,000
/BIC/FZJB_09_10	653	55,000,000	180,000,000	280,000,000
/BIC/FZJB_09_11	653	60,000,000	185,000,000	285,000,000
/BIC/FZJB_09_12	653	60,000,000	185,000,000	285,000,000
/BIC/FZJB_10_01	653	55,000,000	180,000,000	280,000,000
/BIC/FZJB_10_02	653	60,000,000	185,000,000	280,000,000

Table name	Average row length	Total rows 5 TB	Total rows 15.3 TB	Total rows 25 TB
/BIC/FZJB_10_03	653	60,000,000	185,000,000	285,000,000
/BIC/FZJB_10_04	653	50,000,000	175,000,000	275,000,000
/BIC/FZJB_10_05	653	60,000,000	185,000,000	285,000,000
/BIC/FZJB_10_06	653	60,000,000	185,000,000	285,000,000
/BIC/FZJB_10_07	653	60,000,000	185,000,000	285,000,000
/BIC/FZJB_10_08	653	60,000,000	186,631,315	285,000,000
/BIC/FZJB_10_09	653	60,000,000	180,000,000	280,000,000
/BIC/FZJB_10_10	653	55,000,000	180,000,000	280,000,000
/BIC/FZJB_10_11	653	60,000,000	185,000,000	285,000,000
/BIC/FZJB_10_12	653	60,000,000	185,000,000	285,000,000

Tables 5 through 8 show the settings of our bufferpools for each page size.

Table 5 DB2 bufferpool settings - 4 K pages

Parameters	BP0	BP1	BP2	BP3	BP40
VIRTUAL POOL SIZE (in 1000)	4	320	160	160	20
VIRTUAL POOL SEQUENTIAL THRESHOLD	50	100	50	40	50
HORIZONTAL DEFERRED WRITE THRESHOLD	50	50	50	30	50
VERTICAL DEFERRED WRITE THRESHOLD (%)	10	10	5	5	10
VERTICAL DEFERRED WRITE THRESHOLD (BUFFERS)	0	0	0	0	0
VIRTUAL POOL PARALLEL SEQUENTIAL THRESHOLD	50	50	50	50	50
PGFIX ATTRIBUTE	YES	YES	YES	YES	YES
PAGE STEAL METHOD	LRU	LRU	LRU	LRU	LRU

Table 6 DB2 bufferpool settings - 8 k pages

Parameters	BP8K0	BP8K1
VIRTUAL POOL SIZE (in thousands)	160	10
VIRTUAL POOL SEQUENTIAL THRESHOLD	50	80
HORIZONTAL DEFERRED WRITE THRESHOLD	30	30
VERTICAL DEFERRED WRITE THRESHOLD (%)	5	5
VERTICAL DEFERRED WRITE THRESHOLD (BUFFERS)	0	0
VIRTUAL POOL PARALLEL SEQUENTIAL THRESHOLD	50	50
PGFIX ATTRIBUTE	YES	NO
PAGE STEAL METHOD	LRU	LRU

Table 7 DB2 bufferpool settings - 16 k pages

Parameters	BPK16K0	BPK16K1
VIRTUAL POOL SIZE (in thousands)	3	10
VIRTUAL POOL SEQUENTIAL THRESHOLD	50	80
HORIZONTAL DEFERRED WRITE THRESHOLD	30	30
VERTICAL DEFERRED WRITE THRESHOLD (%)	5	5
VERTICAL DEFERRED WRITE THRESHOLD (BUFFERS)	0	0
VIRTUAL POOL PARALLEL SEQUENTIAL THRESHOLD	50	50
PGFIX ATTRIBUTE	YES	NO
PAGE STEAL METHOD	LRU	LRU

Table 8 DB2 bufferpool settings - 32 k pages

Parameters	BP32K	BP32K1	BP32K2	BP32K3	BP32K4
VIRTUAL POOL SIZE (in thousands)	320	320	320	320	80
VIRTUAL POOL SEQUENTIAL THRESHOLD	50	50	50	80	80
HORIZONTAL DEFERRED WRITE THRESHOLD	30	30	30	30	30
VERTICAL DEFERRED WRITE THRESHOLD (%)	5	5	5	5	5
VERTICAL DEFERRED WRITE THRESHOLD (BUFFERS)	0	0	0	0	0
VIRTUAL POOL PARALLEL SEQUENTIAL THRESHOLD	50	50	50	50	50
PGFIX ATTRIBUTE	YES	YES	YES	YES	YES
PAGE STEAL METHOD	LRU	LRU	LRU	LRU	LRU

## BIA index creation

Examples 4 through 8 show cubes that have been indexed to GPFS. They display examples of horizontal partitioning, vertical decomposition, and smart compression of BIA index creation. Example 4 shows a list of cubes that were indexed to BIA blade GPFS.

### Example 4 Cubes indexed to GPFS

```
t11adm@jup001:/gpfs/fs1/T11/TRX00/index> ls -rtl
total 454
drwxr-xr-x  3 t11adm sapsys  2048 2007-08-10 11:52 nap_jsdblc00
drwxr-xr-x 16 t11adm sapsys  2048 2007-08-10 11:53 nap_b49
drwxr-xr-x  3 t11adm sapsys  2048 2007-08-10 11:53 nap_zjb_00_00
drwxr-xr-x  3 t11adm sapsys  2048 2007-08-10 11:58 nap_jsddl00
drwxr-xr-x 139 t11adm sapsys  8192 2007-08-10 12:01 _trex
drwxr-xr-x 108 t11adm sapsys  4096 2007-08-10 12:02 nap_bi0
drwxr-xr-x  3 t11adm sapsys  2048 2007-08-10 12:02 nap_jsdorc00
drwxr-xr-x  3 t11adm sapsys  2048 2007-08-10 12:18 nap_jsdblc01
drwxr-xr-x  3 t11adm sapsys  2048 2007-08-10 12:20 nap_jsddl01
```



```
drwxr-xr-x 3 t11adm sapsys 2048 2007-08-10 13:47 nap_jsdbl06
drwxr-xr-x 3 t11adm sapsys 2048 2007-08-10 13:48 nap_jsdorc05
drwxr-xr-x 3 t11adm sapsys 2048 2007-08-10 13:50 nap_jsdorc06
drwxr-xr-x 3 t11adm sapsys 2048 2007-08-10 13:50 nap_jsddl06
drwxr-xr-x 3 t11adm sapsys 2048 2007-08-10 13:50 nap_jsdbl07
drwxr-xr-x 3 t11adm sapsys 2048 2007-08-10 13:51 nap_jsddl07
drwxr-xr-x 3 t11adm sapsys 2048 2007-08-10 13:53 nap_jsdorc07
drwxr-xr-x 3 t11adm sapsys 2048 2007-08-10 13:58 nap_jsdbl08
drwxr-xr-x 3 t11adm sapsys 2048 2007-08-10 13:59 nap_jsddl08
drwxr-xr-x 3 t11adm sapsys 2048 2007-08-10 14:00 nap_jsdbl09
drwxr-xr-x 3 t11adm sapsys 2048 2007-08-10 14:00 nap_jsdorc08
drwxr-xr-x 3 t11adm sapsys 2048 2007-08-10 14:01 nap_jsddl09
drwxr-xr-x 3 t11adm sapsys 2048 2007-08-10 14:03 nap_jsdorc09
drwxr-xr-x 3 t11adm sapsys 2048 2007-08-10 14:09 nap_jsdbl10
drwxr-xr-x 3 t11adm sapsys 2048 2007-08-10 14:11 nap_jsddl10
drwxr-xr-x 3 t11adm sapsys 2048 2007-08-10 14:11 nap_phsdbl01
drwxr-xr-x 3 t11adm sapsys 2048 2007-08-10 14:11 nap_phsdbl02
drwxr-xr-x 3 t11adm sapsys 2048 2007-08-10 14:14 nap_jsdorc10
drwxr-xr-x 4585 t11adm sapsys 262144 2007-08-10 15:28 nap_bic
t11adm@jup001:/gpfs/fs1/T11/TRX00/index>
```

---

Figure 12 shows TREX Admin settings for horizontal partitioning and the threshold. The threshold was set to max\_size=250,000,000 cells to be considered for horizontal partitioning and max\_split\_parts=40 for number of parts or blade parallelism.

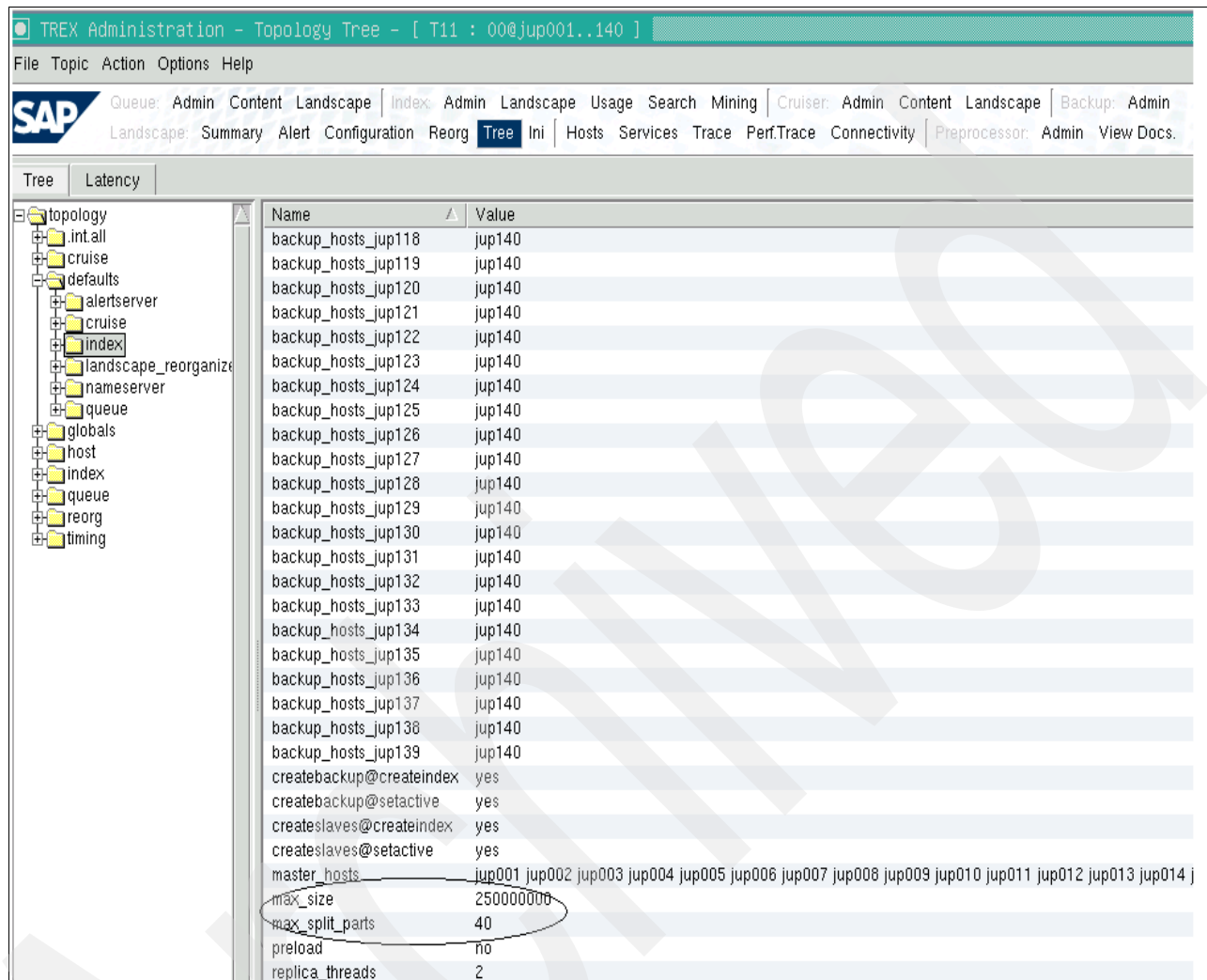


Figure 12 TREX Admin settings for horizontal partitioning and threshold

Example 5 shows a cube index creation with horizontal partitioning of 40 parts.

**Example 5 Horizontal partitioning with 40 parts**

```
t1ladm@jup001:/gpfs/fs1/T11/TRX00/index/nap_bic>ls -rtl
...
drwxr-xr-x 3 t1ladm sapsys 2048 2007-12-05 17:13 fzjb_07_11
drwxr-xr-x 4 t1ladm sapsys 2048 2007-12-05 17:13 fzjb_07_11~01
drwxr-xr-x 4 t1ladm sapsys 2048 2007-12-05 17:13 fzjb_07_11~02
drwxr-xr-x 4 t1ladm sapsys 2048 2007-12-05 17:13 fzjb_07_11~03
drwxr-xr-x 4 t1ladm sapsys 2048 2007-12-05 17:13 fzjb_07_11~04
drwxr-xr-x 4 t1ladm sapsys 2048 2007-12-05 17:13 fzjb_07_11~05
drwxr-xr-x 4 t1ladm sapsys 2048 2007-12-05 17:13 fzjb_07_11~06
drwxr-xr-x 4 t1ladm sapsys 2048 2007-12-05 17:13 fzjb_07_11~07
drwxr-xr-x 4 t1ladm sapsys 2048 2007-12-05 17:13 fzjb_07_11~08
drwxr-xr-x 4 t1ladm sapsys 2048 2007-12-05 17:13 fzjb_07_11~09
drwxr-xr-x 4 t1ladm sapsys 2048 2007-12-05 17:13 fzjb_07_11~10
drwxr-xr-x 4 t1ladm sapsys 2048 2007-12-05 17:13 fzjb_07_11~11
drwxr-xr-x 4 t1ladm sapsys 2048 2007-12-05 17:13 fzjb_07_11~12
```

```

drwxr-xr-x 4 t1ladm sapsys 2048 2007-12-05 17:13 fzjb_07_11~13
drwxr-xr-x 4 t1ladm sapsys 2048 2007-12-05 17:13 fzjb_07_11~14
drwxr-xr-x 4 t1ladm sapsys 2048 2007-12-05 17:13 fzjb_07_11~15
drwxr-xr-x 4 t1ladm sapsys 2048 2007-12-05 17:13 fzjb_07_11~16
drwxr-xr-x 4 t1ladm sapsys 2048 2007-12-05 17:13 fzjb_07_11~17
drwxr-xr-x 4 t1ladm sapsys 2048 2007-12-05 17:13 fzjb_07_11~18
drwxr-xr-x 4 t1ladm sapsys 2048 2007-12-05 17:13 fzjb_07_11~19
drwxr-xr-x 4 t1ladm sapsys 2048 2007-12-05 17:13 fzjb_07_11~20
drwxr-xr-x 4 t1ladm sapsys 2048 2007-12-05 17:13 fzjb_07_11~21
drwxr-xr-x 4 t1ladm sapsys 2048 2007-12-05 17:13 fzjb_07_11~22
drwxr-xr-x 4 t1ladm sapsys 2048 2007-12-05 17:13 fzjb_07_11~23
drwxr-xr-x 4 t1ladm sapsys 2048 2007-12-05 17:13 fzjb_07_11~24
drwxr-xr-x 4 t1ladm sapsys 2048 2007-12-05 17:13 fzjb_07_11~25
drwxr-xr-x 4 t1ladm sapsys 2048 2007-12-05 17:13 fzjb_07_11~26
drwxr-xr-x 4 t1ladm sapsys 2048 2007-12-05 17:14 fzjb_07_11~27
drwxr-xr-x 4 t1ladm sapsys 2048 2007-12-05 17:14 fzjb_07_11~28
drwxr-xr-x 4 t1ladm sapsys 2048 2007-12-05 17:14 fzjb_07_11~29
drwxr-xr-x 4 t1ladm sapsys 2048 2007-12-05 17:14 fzjb_07_11~30
drwxr-xr-x 4 t1ladm sapsys 2048 2007-12-05 17:14 fzjb_07_11~31
drwxr-xr-x 4 t1ladm sapsys 2048 2007-12-05 17:14 fzjb_07_11~32
drwxr-xr-x 4 t1ladm sapsys 2048 2007-12-05 17:14 fzjb_07_11~33
drwxr-xr-x 4 t1ladm sapsys 2048 2007-12-05 17:14 fzjb_07_11~34
drwxr-xr-x 4 t1ladm sapsys 2048 2007-12-05 17:14 fzjb_07_11~35
drwxr-xr-x 4 t1ladm sapsys 2048 2007-12-05 17:14 fzjb_07_11~36
drwxr-xr-x 4 t1ladm sapsys 2048 2007-12-05 17:14 fzjb_07_11~37
drwxr-xr-x 4 t1ladm sapsys 2048 2007-12-05 17:14 fzjb_07_11~38
drwxr-xr-x 4 t1ladm sapsys 2048 2007-12-05 17:14 fzjb_07_11~39
drwxr-xr-x 4 t1ladm sapsys 2048 2007-12-05 17:14 fzjb_07_11~40
drwxr-xr-x 4 t1ladm sapsys 2048 2007-12-05 17:14 dzjb_07_11p
drwxr-xr-x 4 t1ladm sapsys 2048 2007-12-05 17:14 dzjb_07_11u
drwxr-xr-x 4 t1ladm sapsys 2048 2007-12-05 17:14 dzjb_07_11i
drwxr-xr-x 4 t1ladm sapsys 2048 2007-12-05 17:14 dzjb_07_11t
drwxr-xr-x 4 t1ladm sapsys 2048 2007-12-05 17:14 dzjb_07_113
drwxr-xr-x 4 t1ladm sapsys 2048 2007-12-05 17:14 dzjb_07_115
drwxr-xr-x 4 t1ladm sapsys 2048 2007-12-05 17:14 dzjb_07_114
drwxr-xr-x 4 t1ladm sapsys 2048 2007-12-05 17:14 dzjb_07

```

Example 6 displays all the columns in Table SAPR3./BIC/FZJB\_07\_11 using the DB2 Admin tool. It is helpful to observe the column names and match them with the list of temporary and persistent objects in subsequent examples.

*Example 6 Displaying the columns of a fact table of an index cube*

```
DB2 Admin -- DSN9 Columns in Table SAPR3./BIC/FZJB_07_11 > PAGE
```

Select	Column Name	Col	No	Col	Type	Length	Scale	Null	Def	FP	Col	Card
*		*	*			*		*	*	*	*	*
	KEY_ZJB_07_11P	1			INTEGER	4		0	N	4	N	35
	KEY_ZJB_07_11T	2			INTEGER	4		0	N	4	N	30
	KEY_ZJB_07_11U	3			INTEGER	4		0	N	4	N	1
	KEY_ZJB_07_111	4			INTEGER	4		0	N	4	N	0
	KEY_ZJB_07_112	5			INTEGER	4		0	N	4	N	91136
	KEY_ZJB_07_113	6			INTEGER	4		0	N	4	N	81920
	KEY_ZJB_07_114	7			INTEGER	4		0	N	4	N	99
	KEY_ZJB_07_115	8			INTEGER	4		0	N	4	N	99
	SID_OCALMONTH	9			INTEGER	4		0	N	4	N	-1
	/B49/S_CRMEM_CST	10			DECIMAL	17		2	N	4	N	-1
	/B49/S_CRMEM_QTY	11			DECIMAL	17		3	N	4	N	-1
	/B49/S_CRMEM_VAL	12			DECIMAL	17		2	N	4	N	-1

/B49/S_INCORDCST	13	DECIMAL	17	2	N	4	N	-1
/B49/S_INCORDQTY	14	DECIMAL	17	3	N	4	N	-1
/B49/S_INCORDVAL	15	DECIMAL	17	2	N	4	N	-1
/B49/S_INVCD_CST	16	DECIMAL	17	2	N	4	N	-1
/B49/S_INVCD_QTY	17	DECIMAL	17	3	N	4	N	-1
/B49/S_INVCD_VAL	18	DECIMAL	17	2	N	4	N	-1
/B49/S_OPORDQTYB	19	DECIMAL	17	3	N	4	N	-1
/B49/S_OPORDVALS	20	DECIMAL	17	2	N	4	N	-1
/B49/S_ORD_ITEMS	21	DECIMAL	17	3	N	4	N	-1
/B49/S_RTNSCST	22	DECIMAL	17	2	N	4	N	-1
/B49/S_RTNSQTY	23	DECIMAL	17	3	N	4	N	-1
/B49/S_RTNSVAL	24	DECIMAL	17	2	N	4	N	-1
/B49/S_RTNS_ITEM	25	DECIMAL	17	3	N	4	N	-1
/BIC/AMOUNT01	26	DECIMAL	17	2	N	4	N	-1
/BIC/AMOUNT02	27	DECIMAL	17	2	N	4	N	-1
/BIC/AMOUNT03	28	DECIMAL	17	2	N	4	N	-1
/BIC/AMOUNT04	29	DECIMAL	17	2	N	4	N	-1
/BIC/AMOUNT05	30	DECIMAL	17	2	N	4	N	-1
/BIC/AMOUNT06	31	DECIMAL	17	2	N	4	N	-1
/BIC/AMOUNT07	32	DECIMAL	17	2	N	4	N	-1
/BIC/AMOUNT08	33	DECIMAL	17	2	N	4	N	-1
/BIC/AMOUNT09	34	DECIMAL	17	2	N	4	N	-1
/BIC/AMOUNT10	35	DECIMAL	17	2	N	4	N	-1
/BIC/AMOUNT11	36	DECIMAL	17	2	N	4	N	-1
/BIC/AMOUNT12	37	DECIMAL	17	2	N	4	N	-1
/BIC/AMOUNT13	38	DECIMAL	17	2	N	4	N	-1
/BIC/AMOUNT14	39	DECIMAL	17	2	N	4	N	-1
/BIC/AMOUNT15	40	DECIMAL	17	2	N	4	N	-1
/BIC/COUNTER01	41	INTEGER	4	0	N	4	N	-1
/BIC/COUNTER02	42	INTEGER	4	0	N	4	N	-1
/BIC/COUNTER03	43	INTEGER	4	0	N	4	N	-1
/BIC/COUNTER04	44	INTEGER	4	0	N	4	N	-1
/BIC/COUNTER05	45	INTEGER	4	0	N	4	N	-1
/BIC/COUNTER06	46	INTEGER	4	0	N	4	N	-1
/BIC/COUNTER07	47	INTEGER	4	0	N	4	N	-1
/BIC/COUNTER08	48	INTEGER	4	0	N	4	N	-1
/BIC/COUNTER09	49	INTEGER	4	0	N	4	N	-1
/BIC/COUNTER10	50	INTEGER	4	0	N	4	N	-1
/BIC/COUNTER11	51	INTEGER	4	0	N	4	N	-1
/BIC/COUNTER12	52	INTEGER	4	0	N	4	N	-1
/BIC/COUNTER13	53	INTEGER	4	0	N	4	N	-1
/BIC/COUNTER14	54	INTEGER	4	0	N	4	N	-1
/BIC/COUNTER15	55	INTEGER	4	0	N	4	N	-1
/BIC/NUMBER01	56	DECIMAL	17	3	N	4	N	-1
/BIC/NUMBER02	57	DECIMAL	17	3	N	4	N	-1
/BIC/NUMBER03	58	DECIMAL	17	3	N	4	N	-1
/BIC/NUMBER04	59	DECIMAL	17	3	N	4	N	-1
/BIC/NUMBER05	60	DECIMAL	17	3	N	4	N	-1
/BIC/NUMBER06	61	DECIMAL	17	3	N	4	N	-1
/BIC/NUMBER07	62	DECIMAL	17	3	N	4	N	-1
/BIC/NUMBER08	63	DECIMAL	17	3	N	4	N	-1
/BIC/NUMBER09	64	DECIMAL	17	3	N	4	N	-1
/BIC/NUMBER10	65	DECIMAL	17	3	N	4	N	-1
/BIC/NUMBER11	66	DECIMAL	17	3	N	4	N	-1
/BIC/NUMBER12	67	DECIMAL	17	3	N	4	N	-1
/BIC/NUMBER13	68	DECIMAL	17	3	N	4	N	-1
/BIC/NUMBER14	69	DECIMAL	17	3	N	4	N	-1
/BIC/NUMBER15	70	DECIMAL	17	3	N	4	N	-1
/BIC/QUAN01	71	DECIMAL	17	3	N	4	N	-1
/BIC/QUAN02	72	DECIMAL	17	3	N	4	N	-1



/BIC/QUAN03	73	DECIMAL	17	3	N	4	N	-1
/BIC/QUAN04	74	DECIMAL	17	3	N	4	N	-1
/BIC/QUAN05	75	DECIMAL	17	3	N	4	N	-1
/BIC/QUAN06	76	DECIMAL	17	3	N	4	N	-1
/BIC/QUAN07	77	DECIMAL	17	3	N	4	N	-1
/BIC/QUAN08	78	DECIMAL	17	3	N	4	N	-1
/BIC/QUAN09	79	DECIMAL	17	3	N	4	N	-1
/BIC/QUAN10	80	DECIMAL	17	3	N	4	N	-1
/BIC/QUAN11	81	DECIMAL	17	3	N	4	N	-1
/BIC/QUAN12	82	DECIMAL	17	3	N	4	N	-1
/BIC/QUAN13	83	DECIMAL	17	3	N	4	N	-1
/BIC/QUAN14	84	DECIMAL	17	3	N	4	N	-1
/BIC/QUAN15	85	DECIMAL	17	3	N	4	N	-1

Example 7 displays temporary objects of an InfoCube during the first phase of vertical decomposition. The temporary objects will be removed after persistent objects are created. Note that each \*.tmpx file corresponds to a column listed earlier in Example 5

*Example 7 Temporary objects of an indexed cube*

```
t1ladm@jup001:/home/t1ladm/nap_bic/fzjb_07_11~23/en> ls -al
(when it is using local disk)
total 193876
drwxr-xr-x 4 t1ladm sapsys 4096 2007-08-02 09:33 .
drwxr-xr-x 4 t1ladm sapsys 4096 2007-08-02 09:32 ..
drwxr-xr-x 2 t1ladm sapsys 4096 2007-08-02 09:32 attributes
-rw-r--r-- 1 t1ladm sapsys 2007641 2007-08-02 09:44 $b49$s_crmem_cst.tmpx
-rw-r--r-- 1 t1ladm sapsys 2084482 2007-08-02 09:44 $b49$s_crmem_qty.tmpx
-rw-r--r-- 1 t1ladm sapsys 2122285 2007-08-02 09:44 $b49$s_crmem_val.tmpx
-rw-r--r-- 1 t1ladm sapsys 3573234 2007-08-02 09:44 $b49$s_incordcst.tmpx
-rw-r--r-- 1 t1ladm sapsys 2716452 2007-08-02 09:44 $b49$s_incordqty.tmpx
-rw-r--r-- 1 t1ladm sapsys 3573234 2007-08-02 09:44 $b49$s_incordval.tmpx
-rw-r--r-- 1 t1ladm sapsys 3573234 2007-08-02 09:44 $b49$s_invcd_cst.tmpx
-rw-r--r-- 1 t1ladm sapsys 2714842 2007-08-02 09:44 $b49$s_invcd_qty.tmpx
-rw-r--r-- 1 t1ladm sapsys 3555548 2007-08-02 09:44 $b49$s_invcd_val.tmpx
-rw-r--r-- 1 t1ladm sapsys 2057188 2007-08-02 09:44 $b49$s_opordqtyb.tmpx
-rw-r--r-- 1 t1ladm sapsys 2149884 2007-08-02 09:44 $b49$s_opordvals.tmpx
-rw-r--r-- 1 t1ladm sapsys 2799927 2007-08-02 09:44 $b49$s_ord_items.tmpx
-rw-r--r-- 1 t1ladm sapsys 2007641 2007-08-02 09:44 $b49$s_rtncst.tmpx
-rw-r--r-- 1 t1ladm sapsys 2007689 2007-08-02 09:44 $b49$s_rtns_item.tmpx
-rw-r--r-- 1 t1ladm sapsys 2084482 2007-08-02 09:44 $b49$s_rtnsqty.tmpx
-rw-r--r-- 1 t1ladm sapsys 2122285 2007-08-02 09:44 $b49$s_rtnsval.tmpx
-rw-r--r-- 1 t1ladm sapsys 2091822 2007-08-02 09:44 $bic$amount01.tmpx
-rw-r--r-- 1 t1ladm sapsys 2119557 2007-08-02 09:44 $bic$amount02.tmpx
-rw-r--r-- 1 t1ladm sapsys 2131350 2007-08-02 09:44 $bic$amount03.tmpx
-rw-r--r-- 1 t1ladm sapsys 2156915 2007-08-02 09:44 $bic$amount04.tmpx
-rw-r--r-- 1 t1ladm sapsys 2154358 2007-08-02 09:44 $bic$amount05.tmpx
-rw-r--r-- 1 t1ladm sapsys 2156747 2007-08-02 09:44 $bic$amount06.tmpx
-rw-r--r-- 1 t1ladm sapsys 2155379 2007-08-02 09:44 $bic$amount07.tmpx
-rw-r--r-- 1 t1ladm sapsys 2157579 2007-08-02 09:44 $bic$amount08.tmpx
-rw-r--r-- 1 t1ladm sapsys 2156158 2007-08-02 09:44 $bic$amount09.tmpx
-rw-r--r-- 1 t1ladm sapsys 2157088 2007-08-02 09:44 $bic$amount10.tmpx
-rw-r--r-- 1 t1ladm sapsys 2183358 2007-08-02 09:44 $bic$amount11.tmpx
-rw-r--r-- 1 t1ladm sapsys 2181383 2007-08-02 09:44 $bic$amount12.tmpx
-rw-r--r-- 1 t1ladm sapsys 2179923 2007-08-02 09:44 $bic$amount13.tmpx
-rw-r--r-- 1 t1ladm sapsys 2179145 2007-08-02 09:44 $bic$amount14.tmpx
-rw-r--r-- 1 t1ladm sapsys 2176888 2007-08-02 09:44 $bic$amount15.tmpx
-rw-r--r-- 1 t1ladm sapsys 1995904 2007-08-02 09:44 $bic$counter01.tmpx
-rw-r--r-- 1 t1ladm sapsys 1995970 2007-08-02 09:44 $bic$counter02.tmpx
-rw-r--r-- 1 t1ladm sapsys 1995594 2007-08-02 09:44 $bic$counter03.tmpx
```

```

-rw-r--r-- 1 tlladm sapsys 1995658 2007-08-02 09:44 $bic$counter04.tmpx
-rw-r--r-- 1 tlladm sapsys 1995793 2007-08-02 09:44 $bic$counter05.tmpx
-rw-r--r-- 1 tlladm sapsys 1996173 2007-08-02 09:44 $bic$counter06.tmpx
-rw-r--r-- 1 tlladm sapsys 1995748 2007-08-02 09:44 $bic$counter07.tmpx
-rw-r--r-- 1 tlladm sapsys 1996012 2007-08-02 09:44 $bic$counter08.tmpx
-rw-r--r-- 1 tlladm sapsys 1996284 2007-08-02 09:44 $bic$counter09.tmpx
-rw-r--r-- 1 tlladm sapsys 1996173 2007-08-02 09:44 $bic$counter10.tmpx
-rw-r--r-- 1 tlladm sapsys 1995566 2007-08-02 09:44 $bic$counter11.tmpx
-rw-r--r-- 1 tlladm sapsys 1996104 2007-08-02 09:44 $bic$counter12.tmpx
-rw-r--r-- 1 tlladm sapsys 1995707 2007-08-02 09:44 $bic$counter13.tmpx
-rw-r--r-- 1 tlladm sapsys 1996117 2007-08-02 09:44 $bic$counter14.tmpx
-rw-r--r-- 1 tlladm sapsys 1995722 2007-08-02 09:44 $bic$counter15.tmpx
-rw-r--r-- 1 tlladm sapsys 2094322 2007-08-02 09:44 $bic$number01.tmpx
-rw-r--r-- 1 tlladm sapsys 2107627 2007-08-02 09:44 $bic$number02.tmpx
-rw-r--r-- 1 tlladm sapsys 2116150 2007-08-02 09:44 $bic$number03.tmpx
-rw-r--r-- 1 tlladm sapsys 2153452 2007-08-02 09:44 $bic$number04.tmpx
-rw-r--r-- 1 tlladm sapsys 2156289 2007-08-02 09:44 $bic$number05.tmpx
-rw-r--r-- 1 tlladm sapsys 2155899 2007-08-02 09:44 $bic$number06.tmpx
-rw-r--r-- 1 tlladm sapsys 2156776 2007-08-02 09:44 $bic$number07.tmpx
-rw-r--r-- 1 tlladm sapsys 2154834 2007-08-02 09:44 $bic$number08.tmpx
-rw-r--r-- 1 tlladm sapsys 2154793 2007-08-02 09:44 $bic$number09.tmpx
-rw-r--r-- 1 tlladm sapsys 2154130 2007-08-02 09:44 $bic$number10.tmpx
-rw-r--r-- 1 tlladm sapsys 2178435 2007-08-02 09:44 $bic$number11.tmpx
-rw-r--r-- 1 tlladm sapsys 2179602 2007-08-02 09:44 $bic$number12.tmpx
-rw-r--r-- 1 tlladm sapsys 2179118 2007-08-02 09:44 $bic$number13.tmpx
-rw-r--r-- 1 tlladm sapsys 2181036 2007-08-02 09:44 $bic$number14.tmpx
-rw-r--r-- 1 tlladm sapsys 2181499 2007-08-02 09:44 $bic$number15.tmpx
-rw-r--r-- 1 tlladm sapsys 2084057 2007-08-02 09:44 $bic$quan01.tmpx
-rw-r--r-- 1 tlladm sapsys 2097352 2007-08-02 09:44 $bic$quan02.tmpx
-rw-r--r-- 1 tlladm sapsys 2104614 2007-08-02 09:44 $bic$quan03.tmpx
-rw-r--r-- 1 tlladm sapsys 2109508 2007-08-02 09:44 $bic$quan04.tmpx
-rw-r--r-- 1 tlladm sapsys 2114724 2007-08-02 09:44 $bic$quan05.tmpx
-rw-r--r-- 1 tlladm sapsys 2155726 2007-08-02 09:44 $bic$quan06.tmpx
-rw-r--r-- 1 tlladm sapsys 2158725 2007-08-02 09:44 $bic$quan07.tmpx
-rw-r--r-- 1 tlladm sapsys 2157700 2007-08-02 09:44 $bic$quan08.tmpx
-rw-r--r-- 1 tlladm sapsys 2156676 2007-08-02 09:44 $bic$quan09.tmpx
-rw-r--r-- 1 tlladm sapsys 2157039 2007-08-02 09:44 $bic$quan10.tmpx
-rw-r--r-- 1 tlladm sapsys 2155510 2007-08-02 09:44 $bic$quan11.tmpx
-rw-r--r-- 1 tlladm sapsys 2154730 2007-08-02 09:44 $bic$quan12.tmpx
-rw-r--r-- 1 tlladm sapsys 2155985 2007-08-02 09:44 $bic$quan13.tmpx
-rw-r--r-- 1 tlladm sapsys 2155565 2007-08-02 09:44 $bic$quan14.tmpx
-rw-r--r-- 1 tlladm sapsys 2156150 2007-08-02 09:44 $bic$quan15.tmpx
-rw-r--r-- 1 tlladm sapsys 5425676 2007-08-02 09:44 key_zjb_07_111.tmpx
-rw-r--r-- 1 tlladm sapsys 6091352 2007-08-02 09:44 key_zjb_07_112.tmpx
-rw-r--r-- 1 tlladm sapsys 5935488 2007-08-02 09:44 key_zjb_07_113.tmpx
-rw-r--r-- 1 tlladm sapsys 3040371 2007-08-02 09:44 key_zjb_07_114.tmpx
-rw-r--r-- 1 tlladm sapsys 3555863 2007-08-02 09:44 key_zjb_07_115.tmpx
-rw-r--r-- 1 tlladm sapsys 1950018 2007-08-02 09:44 key_zjb_07_11p.tmpx
-rw-r--r-- 1 tlladm sapsys 2182508 2007-08-02 09:44 key_zjb_07_11t.tmpx
-rw-r--r-- 1 tlladm sapsys 1949503 2007-08-02 09:44 key_zjb_07_11u.tmpx
drwxr-xr-x 2 tlladm sapsys 4096 2007-08-02 09:32 trex

```

---

Figure 8 shows permanent or persistent objects with files \*.bin as final vertical decomposition. Again, each \*.bin file corresponds to a column listed in Example 5. Also notice the file size differences between \*.bin and \*.tmpx files, after compression.

*Example 8 Permanent objects of an indexed cube*

```
t1ladm@jup001:/gpfs/fs1/T11/TRX00/index/nap_bic/fzjb_07_11~23/en/attributes> ls -rtl
total 13874
-rw-r--r-- 1 t1ladm sapsys      12 2007-08-10 12:37 iattribute_$termfrequencies$.bin
-rw-r--r-- 1 t1ladm sapsys 196000 2007-08-10 12:45 attribute_key_zjb_07_11p.bin
-rw-r--r-- 1 t1ladm sapsys 390756 2007-08-10 12:45 attribute_key_zjb_07_11t.bin
-rw-r--r-- 1 t1ladm sapsys      62 2007-08-10 12:45 attribute_key_zjb_07_11u.bin
-rw-r--r-- 1 t1ladm sapsys 310897 2007-08-10 12:45 attribute_$b49$s_ord_items.bin
-rw-r--r-- 1 t1ladm sapsys 306584 2007-08-10 12:45 attribute_$b49$s_invcd_qty.bin
-rw-r--r-- 1 t1ladm sapsys  52287 2007-08-10 12:45 attribute_$b49$s_rtns_item.bin
-rw-r--r-- 1 t1ladm sapsys 536801 2007-08-10 12:45 attribute_$b49$s_invcd_val.bin
-rw-r--r-- 1 t1ladm sapsys  84121 2007-08-10 12:45 attribute_$b49$s_opordvals.bin
-rw-r--r-- 1 t1ladm sapsys 547449 2007-08-10 12:45 attribute_$b49$s_invcd_cst.bin
-rw-r--r-- 1 t1ladm sapsys  64610 2007-08-10 12:45 attribute_$bic$number01.bin
-rw-r--r-- 1 t1ladm sapsys  53914 2007-08-10 12:45 attribute_$bic$number02.bin
-rw-r--r-- 1 t1ladm sapsys  85588 2007-08-10 12:45 attribute_$bic$number03.bin
-rw-r--r-- 1 t1ladm sapsys  61248 2007-08-10 12:45 attribute_$bic$number04.bin
-rw-r--r-- 1 t1ladm sapsys  64187 2007-08-10 12:45 attribute_$b49$s_crmem_qty.bin
-rw-r--r-- 1 t1ladm sapsys  72452 2007-08-10 12:45 attribute_$bic$number10.bin
-rw-r--r-- 1 t1ladm sapsys 104971 2007-08-10 12:45 attribute_$bic$number05.bin
-rw-r--r-- 1 t1ladm sapsys 120335 2007-08-10 12:45 attribute_$bic$number11.bin
-rw-r--r-- 1 t1ladm sapsys 104392 2007-08-10 12:45 attribute_$bic$number06.bin
-rw-r--r-- 1 t1ladm sapsys  34538 2007-08-10 12:45 attribute_$bic$amount01.bin
-rw-r--r-- 1 t1ladm sapsys 112041 2007-08-10 12:45 attribute_$bic$number12.bin
-rw-r--r-- 1 t1ladm sapsys  89547 2007-08-10 12:45 attribute_$bic$number07.bin
-rw-r--r-- 1 t1ladm sapsys  81451 2007-08-10 12:45 attribute_$bic$amount02.bin
-rw-r--r-- 1 t1ladm sapsys  85326 2007-08-10 12:45 attribute_$bic$number13.bin
-rw-r--r-- 1 t1ladm sapsys  95457 2007-08-10 12:45 attribute_$bic$number08.bin
-rw-r--r-- 1 t1ladm sapsys  75548 2007-08-10 12:45 attribute_$bic$amount03.bin
-rw-r--r-- 1 t1ladm sapsys 117510 2007-08-10 12:45 attribute_$bic$number14.bin
-rw-r--r-- 1 t1ladm sapsys 108196 2007-08-10 12:45 attribute_$bic$number09.bin
-rw-r--r-- 1 t1ladm sapsys  91261 2007-08-10 12:45 attribute_$bic$amount04.bin
-rw-r--r-- 1 t1ladm sapsys 111757 2007-08-10 12:45 attribute_$bic$number15.bin
-rw-r--r-- 1 t1ladm sapsys  85718 2007-08-10 12:45 attribute_$bic$amount10.bin
-rw-r--r-- 1 t1ladm sapsys  57490 2007-08-10 12:45 attribute_$bic$amount05.bin
-rw-r--r-- 1 t1ladm sapsys 111879 2007-08-10 12:45 attribute_$bic$amount11.bin
-rw-r--r-- 1 t1ladm sapsys  84593 2007-08-10 12:45 attribute_$bic$amount06.bin
-rw-r--r-- 1 t1ladm sapsys  78207 2007-08-10 12:45 attribute_$bic$amount12.bin
-rw-r--r-- 1 t1ladm sapsys  60472 2007-08-10 12:45 attribute_$bic$amount07.bin
-rw-r--r-- 1 t1ladm sapsys 102033 2007-08-10 12:45 attribute_$bic$amount13.bin
-rw-r--r-- 1 t1ladm sapsys  64900 2007-08-10 12:45 attribute_$bic$amount08.bin
-rw-r--r-- 1 t1ladm sapsys 126182 2007-08-10 12:45 attribute_$bic$amount14.bin
-rw-r--r-- 1 t1ladm sapsys 101342 2007-08-10 12:45 attribute_$bic$amount09.bin
-rw-r--r-- 1 t1ladm sapsys 114999 2007-08-10 12:45 attribute_$bic$amount15.bin
-rw-r--r-- 1 t1ladm sapsys  78057 2007-08-10 12:45 attribute_$b49$s_crmem_val.bin
-rw-r--r-- 1 t1ladm sapsys  52276 2007-08-10 12:45 attribute_$b49$s_crmem_cst.bin
-rw-r--r-- 1 t1ladm sapsys 1517722 2007-08-10 12:45 attribute_key_zjb_07_111.bin
-rw-r--r-- 1 t1ladm sapsys 1517776 2007-08-10 12:45 attribute_key_zjb_07_112.bin
-rw-r--r-- 1 t1ladm sapsys 1483329 2007-08-10 12:45 attribute_key_zjb_07_113.bin
-rw-r--r-- 1 t1ladm sapsys  547149 2007-08-10 12:45 attribute_key_zjb_07_114.bin
-rw-r--r-- 1 t1ladm sapsys  547149 2007-08-10 12:45 attribute_key_zjb_07_115.bin
-rw-r--r-- 1 t1ladm sapsys  70536 2007-08-10 12:45 attribute_$b49$s_opordqtyb.bin
-rw-r--r-- 1 t1ladm sapsys  32770 2007-08-10 12:45 attribute_$bic$counter01.bin
-rw-r--r-- 1 t1ladm sapsys  10218 2007-08-10 12:45 attribute_$bic$counter02.bin
-rw-r--r-- 1 t1ladm sapsys  56330 2007-08-10 12:45 attribute_$bic$counter03.bin
-rw-r--r-- 1 t1ladm sapsys  12266 2007-08-10 12:45 attribute_$bic$counter04.bin
```

```

-rw-r--r-- 1 t1ladm sapsys 29442 2007-08-10 12:45 attribute_$bic$counter10.bin
-rw-r--r-- 1 t1ladm sapsys 36098 2007-08-10 12:45 attribute_$bic$counter05.bin
-rw-r--r-- 1 t1ladm sapsys 64187 2007-08-10 12:45 attribute_$b49$s_rtnsqty.bin
-rw-r--r-- 1 t1ladm sapsys 312464 2007-08-10 12:45 attribute_$b49$s_incordqty.bin
-rw-r--r-- 1 t1ladm sapsys 24322 2007-08-10 12:45 attribute_$bic$counter11.bin
-rw-r--r-- 1 t1ladm sapsys 56610 2007-08-10 12:45 attribute_$bic$counter06.bin
-rw-r--r-- 1 t1ladm sapsys 58074 2007-08-10 12:45 attribute_$bic$counter12.bin
-rw-r--r-- 1 t1ladm sapsys 6618 2007-08-10 12:45 attribute_$bic$counter07.bin
-rw-r--r-- 1 t1ladm sapsys 13290 2007-08-10 12:45 attribute_$bic$counter13.bin
-rw-r--r-- 1 t1ladm sapsys 52746 2007-08-10 12:45 attribute_$bic$counter08.bin
-rw-r--r-- 1 t1ladm sapsys 15858 2007-08-10 12:45 attribute_$bic$counter14.bin
-rw-r--r-- 1 t1ladm sapsys 57114 2007-08-10 12:45 attribute_$bic$counter09.bin
-rw-r--r-- 1 t1ladm sapsys 4818 2007-08-10 12:45 attribute_$bic$counter15.bin
-rw-r--r-- 1 t1ladm sapsys 56035 2007-08-10 12:45 attribute_$bic$quan01.bin
-rw-r--r-- 1 t1ladm sapsys 33886 2007-08-10 12:45 attribute_$bic$quan02.bin
-rw-r--r-- 1 t1ladm sapsys 67441 2007-08-10 12:45 attribute_$bic$quan03.bin
-rw-r--r-- 1 t1ladm sapsys 57880 2007-08-10 12:45 attribute_$bic$quan04.bin
-rw-r--r-- 1 t1ladm sapsys 81661 2007-08-10 12:45 attribute_$bic$quan10.bin
-rw-r--r-- 1 t1ladm sapsys 35785 2007-08-10 12:45 attribute_$bic$quan05.bin
-rw-r--r-- 1 t1ladm sapsys 77588 2007-08-10 12:45 attribute_$bic$quan11.bin
-rw-r--r-- 1 t1ladm sapsys 101134 2007-08-10 12:45 attribute_$bic$quan06.bin
-rw-r--r-- 1 t1ladm sapsys 102855 2007-08-10 12:45 attribute_$bic$quan12.bin
-rw-r--r-- 1 t1ladm sapsys 87969 2007-08-10 12:45 attribute_$bic$quan07.bin
-rw-r--r-- 1 t1ladm sapsys 63103 2007-08-10 12:45 attribute_$bic$quan13.bin
-rw-r--r-- 1 t1ladm sapsys 74679 2007-08-10 12:45 attribute_$bic$quan08.bin
-rw-r--r-- 1 t1ladm sapsys 98341 2007-08-10 12:45 attribute_$bic$quan14.bin
-rw-r--r-- 1 t1ladm sapsys 51910 2007-08-10 12:45 attribute_$bic$quan09.bin
-rw-r--r-- 1 t1ladm sapsys 87323 2007-08-10 12:45 attribute_$bic$quan15.bin
-rw-r--r-- 1 t1ladm sapsys 78057 2007-08-10 12:45 attribute_$b49$s_rtnsval.bin
-rw-r--r-- 1 t1ladm sapsys 547449 2007-08-10 12:45 attribute_$b49$s_incordval.bin
-rw-r--r-- 1 t1ladm sapsys 52276 2007-08-10 12:45 attribute_$b49$s_rtnscst.bin
-rw-r--r-- 1 t1ladm sapsys 547449 2007-08-10 12:45 attribute_$b49$s_incordcst.bin
-rw-r--r-- 1 t1ladm sapsys 12991 2007-08-10 12:45 AttributeStore.ini
t1ladm@jup00
1:/gpfs/fs1/T11/TRX00/index/nap_bic/fzjb_07_11~23/en/attributes>

```

---

## The team that wrote this IBM Redpaper

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

**Seewah Chan** works in the IBM Systems Development SAP Performance group.

**Leticia Cruz** works in the IBM System z Benchmark Center.

**Veng Ly** works in the IBM SAP/R3 Performance Test group.

**Howard Poole** works in the IBM SAP/R3 Performance Test group.

**Christopher Williams** works in the IBM System z Benchmark Center.

Thanks to the following contributor for helping to publish this project:

**Mike Ebbers** IBM International Technical Support Organization, Poughkeepsie Center.

# Notices

This information was developed for products and services offered in the U.S.A.

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 give you any license to these patents. You can send license inquiries, in writing, to:

*IBM Director of Licensing, IBM Corporation, North Castle Drive, Armonk, NY 10504-1785 U.S.A.*

**The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law:** 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 states 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 Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

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

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.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise 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.

This document REDP-4411-00 was created or updated on July 21, 2008.



Send us your comments in one of the following ways:


- ▶ Use the online **Contact us** review Redbooks form found at:  
[ibm.com/redbooks](http://ibm.com/redbooks)
- ▶ Send your comments in an email to:  
[redbooks@us.ibm.com](mailto:redbooks@us.ibm.com)
- ▶ Mail your comments to:  
IBM Corporation, International Technical Support Organization  
Dept. HYTD Mail Station P099  
2455 South Road  
Poughkeepsie, NY 12601-5400 U.S.A.



## Trademarks

IBM, the IBM logo, and [ibm.com](http://ibm.com) are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both. These and other IBM trademarked terms are marked on their first occurrence in this information with the appropriate symbol (® or ™), indicating 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 <http://www.ibm.com/legal/copytrade.shtml>

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

AIX®	GPFS™	System x™
BladeCenter®	IBM®	System z™
DB2 Connect™	POWER5™	System z9®
DB2®	Redbooks (logo)  ®	z/OS®
DS4000™	System p™	z9™
FICON®	System p5™	
General Parallel File System™	System Storage™	

The following terms are trademarks of other companies:

InfiniBand, and the InfiniBand design marks are trademarks and/or service marks of the InfiniBand Trade Association.

ABAP, SAP NetWeaver, SAP, and SAP logos are trademarks or registered trademarks of SAP AG in Germany and in several other countries.

Intel Xeon, Intel, Intel logo, Intel Inside logo, and Intel Centrino logo are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States, other countries, or both.

Linux is a trademark of Linus Torvalds in the United States, other countries, or both.

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