Redbooks

ibm.com/redbooks

z/OS Infrastructure Optimization using Large Memory

Peter Sutton

IBM Academy of Technology

Learn

IBM z Systems

IBM

Point-of-View

z/OS Infrastructure Optimization using Large Memory

An IBM Redbooks Point-of-View publication from IBM Systems

By Peter Sutton, Distinguished Engineer

Highlights

Here's some of what large memory and the IBM z13 can help you achieve:

- Improve transaction response times. Internally, this raises productivity. Externally, it drives more sales.
- Reduce CPU usage and improve CPU scaling. The former saves money, the latter adds flexibility.
- Simplify memory management. Skills now spent tuning memory can be applied to driving business value.
- Improve availability. Tolerate larger memory and I/O workload spikes. Enable faster recovery.
- Deploy more of today's memory-hungry workloads with IBM z Systems industry-leading security and lower total cost of ownership.

Large memory, big results

Businesses can improve their bottom line by changing the way they approach deploying memory on IBM z/OS®. Advances in z Systems memory packaging and pricing have radically shifted the balance between memory, I/O, and CPU, and now favor using large memory for both existing and new workloads. The new balance point optimizes infrastructure by concentrating on the value achievable with large memory instead of tuning to minimal memory usage.

Getting value from large memory requires a change in approach. Historically, choosing how much memory to use has involved finding the minimum amount of memory needed for z/OS to operate successfully and then adding additional memory as a reserve for unexpected business growth or changing business needs. But what if organizations instead adopt a large memory approach and configure new systems with substantially more memory than the minimum required?

This IBM® Redbooks® Point-of-View publication describes the advantages of configuring z Systems products with large memory, including faster response time, lower CPU consumption, and higher availability. With the release of the IBM z13™, IBM is shifting the memory discussion from "Can I survive this year on my current memory footprint?" to "How can I use additional memory to drive new business value?" From higher system availability and customer satisfaction to increased sales and lower costs, these new large memory configurations have a lot to offer.

Response time is critical in business applications

The value of response time improvements for business applications is clear. Customers are likely to buy more and express greater satisfaction when you present information with fast, consistent response times. Even within a company, employee productivity, including call center productivity, typically goes up as application response times improve.

Response time pressure in data centers

Applications use data from many platforms, with sources both inside and outside the data center. Value is created when applications fully use this available information. But this comes at the expense of the additional response time needed to obtain the information. Single-thread CPU speed improvements have

not provided enough response time improvement to keep up with the explosion in application functionality.

Applications that are deployed in a cloud infrastructure, even a private cloud, face additional response time challenges. A cloud infrastructure, which masks location awareness, typically adds network and queuing delays as requests are serviced within it. Mobile workloads, too, can create their own form of capacity-based response time pressure, especially during spikes where I/O infrastructure becomes more heavily loaded. Even simple tasks, such as transferring files, face elapsed time pressure as the amount of data transferred within files continues to grow at high sustained rates.

Large memory helps response time

When IBM tested the large memory concept by increasing the size of the IBM DB2® database buffers, larger buffers improved the response time for applications by up to 50%.[1] The improvement came from replacing lengthy I/O operations with much faster direct access to data residing in the buffer pools. Figure 1 shows that the average database response time in an application came down sharply as more memory was added.

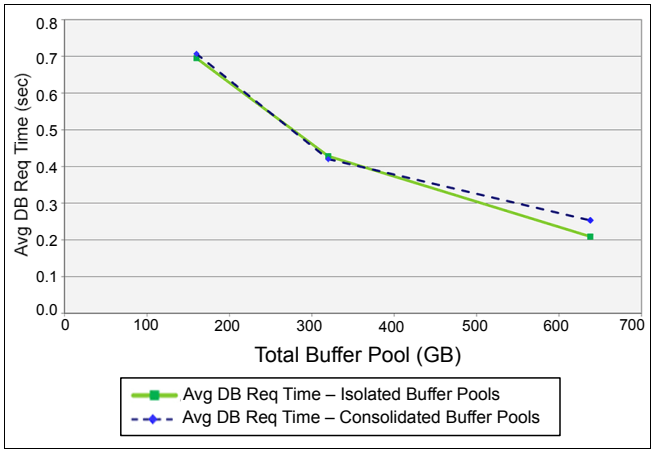

Figure 1 Effect of buffer pool scaling on database response

Even simple file transfer protocols such as FTP can see elapsed time improvements when larger network (TCP) buffers are used.

Work published by the Los Alamos National Laboratory shows that larger TCP buffer sizes can improve TCP throughput by "orders of magnitude over high delay-bandwidth links."[2] In addition, an IBM study using an IBM production TCP network found that using larger TPC/IP memory buffers increased FTP throughput there by up to 1.98 times.[3]

Reducing CPU consumption

Deploying larger DB2 buffer pools and keeping more data in memory can reduce CPU consumption. The CPU benefit comes from removing synchronous read I/O operations (read operations where the application cannot continue processing until the data is available).

Table 1 cites an IBM study[4] showing that the internal throughput rate (ITR), which is the number of transactions the system can run at 100% CPU utilization, actually increases as the size of the DB2 buffer pool increases. This study provides an interesting result: Simply increasing the size of the buffer pool makes the system more efficient, cuts CPU consumption, and potentially reduces software costs. For the tested application, as the buffer pool was increased from 10 GB to 70 GB, without any other changes, total CPU consumption was reduced by an astonishing 23.8%. This extreme performance improvement was driven by a large reduction in synchronous read I/O operations and, therefore, shows the upper limits of possible performance gain. In contrast, workloads with few or no synchronous reads, such as sequential batch workloads, see little or no CPU gain.

Table 1 CPU savings, online database transaction workload

Memory	Buffer	CPU%	ITR	ITR Gain
250 GB	10 GB	40%	12749	NA
250 GB	24 GB	35%	15036	17.9%
250 GB	70 GB	38%	15778	23.8%
250 GB	150 GB	45%	15364	20.5%

[1] *IBM zEnterprise® System: Performance Report on Exploiting Large Memory for DB2 Buffer Pools with SAP:* http://www-03.ibm.com/support/techdocs/atsmastr.nsf/WebIndex/WP102461

[2] *Dynamic Right-Sizing (DRS) Software Distribution:* http://public.lanl.gov/radiant/software/drs.html

[3] z/OS V1R11 Communications Server Performance Summary: http://www-01.ibm.com/support/docview.wss?uid=swg27017239&aid=1

[4] *System z: Advantages of Configuring More Memory for DB2 Buffer Pools:* http://www-03.ibm.com/support/techdocs/atsmastr.nsf/WebIndex/WP102464

IBM expects that typical transactional DB2 workloads will see 0% to 5% CPU gain with larger buffer pools. DB2 v11 comes with a new buffer pool simulation capability you can use to estimate the response time improvements and CPU savings your workloads can achieve with larger buffer pools. Table 1 also shows that the best buffer pool size for this workload (in other words, the size with the highest associated ITR) is 70 GB. Jumping from 70 GB to 160 GB actually reduced the measured CPU gain, which highlights the importance of finding the *correct* buffer pool size.

Large memory CPU savings also come from tuning middleware and z/OS to take advantage of page-fixed memory and 1 MB and 2 GB pages. Using more memory for page-fixed pages allows more efficient I/O operations. 1 MB and 2 GB large pages (LPs) are IBM z Systems architectural extensions that improve CPU performance by making the processor more efficient. The use of page-fixed pages and 1 MB and 2 GB LPs can provide a substantial performance improvement. For example, IBM WebSphere® Day Trader 2.0 benchmarks showed approximately 5% throughput improvement by using 1 MB LPs versus 4 K pages.[5] Figure 2 shows study results for a Java multi-threaded workload. In the figure, the values labeled “Percent ITR improvement” show that the total performance gains from deploying either 1 MB or 2 GB LPs for this workload are also approximately 5%.

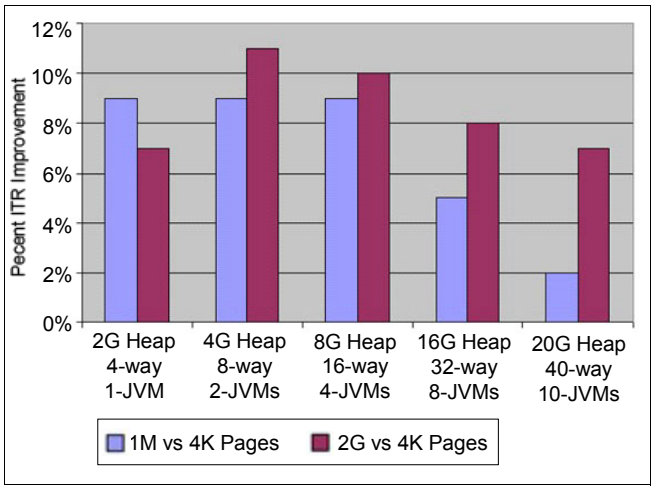

Figure 2 Performance gains for Java multithreaded workload

[5] *Engineered Together—IBM Java and zEC12 Boost Workload Performance, Large Pages Frames:* http://www.ibmsystemsmag.com/mainframe/trends/whatsnew/java_compiler/?page=3

DB2-related CPU consumption can also be reduced through memory tuning. IBM experts have found that DB2 workloads experience up to a 5% CPU benefit from the combination of page-fixed 1 MB pages, thread reuse, and global and local statement caches. For each of these tuning items, more memory is used to gain the CPU reduction. Figure 3 gives an example of DB2 transactions consuming substantially less CPU time when using 1 MB fixed pages instead of 4 K pageable pages.

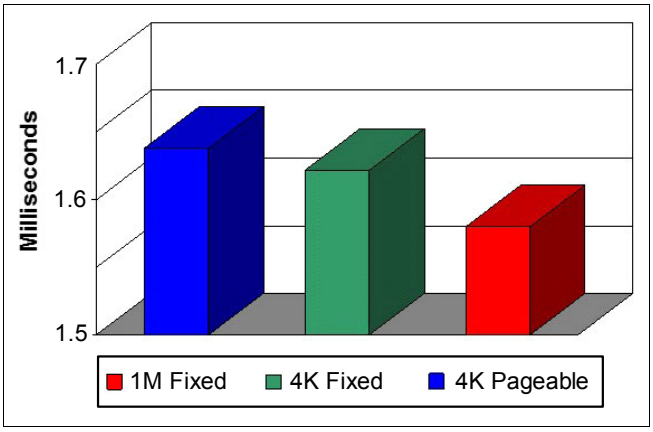

Figure 3 Total DB2 CPU time per transaction

Removing obstacles to workload development and deployment

Workload placement decisions are based in part on the amount of memory consumed by the application. And although z Systems memory is traditionally faster, more reliable, efficiently used, and easily shared than memory on other platforms, small memory configurations (driven by memory cost) have hindered deployment of memory-intensive workloads on z Systems platforms.

The IBM z13 retains the z Systems platform’s established memory advantages, such as high availability RAIM memory, while increasing memory size from 3,000 GB to 10,000 GB and lowering the price. And not surprisingly, more memory, in and of itself, makes memory governance simpler and allows stricter adherence to industry-leading practices for memory use by applications and middleware. Workload placement decisions that are currently restricted by memory footprint size can now be optimized based on total cost of ownership estimates and other business requirements, such as audit cost, disaster recovery

capability, and application availability. For example, creating a large memory environment suitable for analytics processing on z Systems can bring analytics workloads closer to the transactional data targeted for study, resulting in better decision-making based on fresher data that is processed at a potentially lower cost.

Taking the load off the disk I/O network

Many industries have seen extraordinary growth in data volume and consistent growth in I/O access rates. Increasing the capacity of the disk I/O network to meet the demands of data growth typically involves running additional fiber under the floor or through the ceiling of the data center, deploying newer or faster channels, deploying additional or upgraded switches, and adding additional storage devices. Naturally, this process is expensive and requires careful planning so as not to disrupt running workloads.

Deploying DB2 large memory database buffers and application caches can reduce the frequency of disk I/O operations and reduce the load on the disk I/O network. For example, in the response time study referred to earlier in this document, IBM found that increasing the DB2 buffer size eliminated 38,000 I/O operations per second in an SAP Banking Services (SBS) Day Posting workload. Your specific I/O savings from large DB2 buffer pools can be projected using the buffer pool simulation tool available in DB2 V11. The I/O reduction from larger DB2 buffer pools also helps to mitigate workload spikes, especially the mostly read-only query traffic spikes typical of mobile workloads.

Improving availability

Deploying large memory can help maintain z System availability and mitigate the duration of some service interruptions. z/OS systems are typically configured with enough memory to survive missing interrupt and I/O device recovery times, including disk hyperswap. Adding more memory for z/OS system use extends the time before z/OS is forced to start aggressive paging, and limits the amount of rejected network traffic that is externally visible. Minimal paging substantially reduces the time needed for z/OS to recover to full performance following recovery actions.

Businesses can also use large memory to simplify management of memory-sensitive tuning parameters to improve availability. A wide range of parameters are available to enable the use of minimal memory configurations. Adjusting these parameters as systems grow is expensive. Failing to adjust these parameters can be even more costly. Using large memory to allow more relaxed tuning can improve system availability and simplify subsystem tuning. Skills that were spent controlling memory footprints can instead be used to drive new business value.

What's next: How IBM can help

The IBM z13 offers a substantial memory size increase compared to the earlier z196 and zEC12 systems and offers new pricing for increasing memory size. Together, these changes can make migrating to the IBM z13 a good option for gaining the value of large memory.

There are guides that can help your performance team tune your z/OS system and the IBM middleware products that work on z/OS to take advantage of large memory. There also are published IBM Redbooks and IBM Redpaper™ publications on system tuning.

IBM also offers z Systems Large Memory Assessment Services to help you get the most value from large memory.

For more information about these and other resources, see the next section.

Resources for more information

For additional information about how using large memory with the IBM z13 can benefit your organization, refer to these resources:

- DB2 11 for z/OS

  http://www-01.ibm.com/support/knowledgecenter/SSEPEK_11.0.0/com.ibm.db2z11.doc/src/alltoc/db2z_11_prodhome.dita

- *DB2 10 for z/OS Performance Topics*, SG24-7942

  http://www.redbooks.ibm.com/abstracts/sg247942.html?Open

- *15142: z/OS Large Memory: Size Does Matter*

  https://share.confex.com/share/122/webprogram/Session15142.html

- *System z: Advantages of Configuring More Memory for DB2 Buffer Pools*

  http://www-03.ibm.com/support/techdocs/atsmastr.nsf/WebIndex/WP102464

- *Engineered Together-IBM Java and zEC12 Boost Workload Performance, Large Pages Frames*

  http://www.ibmsystemsmag.com/mainframe/trends/whatsnew/java_compiler/?page=3

- *IBM zEnterprise System: Performance Report on Exploiting Large Memory for DB2 Buffer Pools with SAP*

  http://www-03.ibm.com/support/techdocs/atsmastr.nsf/WebIndex/WP102461

- z/OS V1R11 Communications Server Performance Summary

  http://www-01.ibm.com/support/docview.wss?uid=swg27017239&aid=1

- IBM z Systems Large Memory Assessment Services

  http://www-03.ibm.com/systems/services/

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.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

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-5146-00, was created or updated on February 5, 2015.

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. 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:
DB2®
IBM®
Redbooks®
Redpaper™
Redbooks (logo) 
WebSphere®
z/OS®
zEnterprise®

The following terms are trademarks of other companies:

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

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