Redbooks
ibm.com/redbooks

z Systems Simultaneous Multithreading Revolution

Daniel Rosa

Donald Schmidt

Romney White

IBM Academy of Technology

Learn

IBM z Systems

IBM

Point-of-View

z Systems Simultaneous Multithreading Revolution

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

By **Daniel Rosa**, Senior Software Engineer,
Donald Schmidt, Distinguished Engineer, and
Romney White, Senior Technical Staff Member

Highlights

- Typical simultaneous multithreading (SMT) implementations allow workloads from different control programs to use a compute core concurrently, but with variable core capacity gains and thread execution slowdowns.
- The new IBM z13 platform implements SMT differently. At any point in time, a single control program manages the entire core, giving each workload more repeatable core capacity for processing.
- The z13 has new instrumentation that helps control programs deliver real-time measurements of the SMT-based allocations of core resources.
- The z/OS and z/VM control programs use SMT on the z13 to optimize their workloads while providing repeatable metrics for capacity planning and chargeback.

Cloud and SMT in the IT industry

The cloud hungers for compute core capacity to serve dynamic workloads for client hypervisors and operating systems (referred to here as control programs) and their applications. Yet in data-oriented cloud workloads, cache misses often create conditions where a core has no ready instructions to run, preventing it from achieving its maximum throughput and effectively reducing its capacity. Throughput for each workload can vary significantly based on its characteristics, including the frequency and duration of any cache misses it causes.

With simultaneous multithreading (SMT), multiple threads inject instructions into the same core concurrently to increase the likelihood that the core has a ready instruction to run. This increases core throughput and therefore core capacity (see Figure 1). When all threads on the core are injecting instructions from a workload, the core runs at its maximum capacity for that workload. When only some of the core's threads are injecting instructions for the workload, part of the core's capacity is in use and the remainder is free.

Other factors also affect SMT variability. When multiple threads have ready instructions, they share the core's resources, which causes individual thread execution to slow down. This is shown in Figure 1, which shows two workloads sharing the core equally without SMT, whereas with SMT (and two threads per core), each workload runs on one thread and receives the same throughput but in less elapsed time, leaving some capacity free. The number of threads per core and the characteristics of the different workloads (including the frequency and duration of cache misses) are the main contributors to SMT variability and therefore the main influencers of core throughput, effective core capacity, and thread speed.

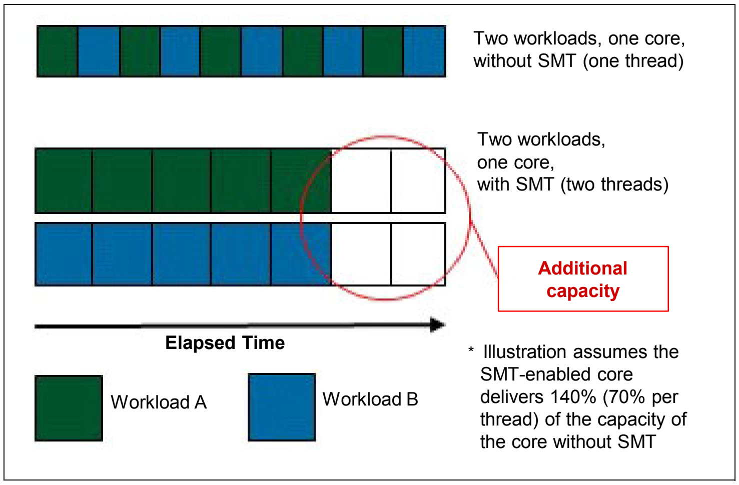
Two workloads, one core, without SMT (one thread)
Two workloads, one core, with SMT (two threads)
Additional capacity
Elapsed Time
Workload A
Workload B
* Illustration assumes the SMT-enabled core delivers 140% (70% per thread) of the capacity of the core without SMT

Figure 1 Core execution without and with SMT

In cloud infrastructures, SMT can be implemented with SMT-aware hardware and hypervisor layers, where the hypervisor provides SMT transparently to SMT-unaware client control programs. With two threads per core, the hypervisor can pair any two control program CPUs on a core (see Figure 2). As the cloud serves more client control programs, the number of workloads (and the number of potential pairs of candidate CPUs on a single core) also increases. These aspects of SMT design exacerbate its variability in all layers because different pairings result in different core throughput, core capacity, and thread speed.

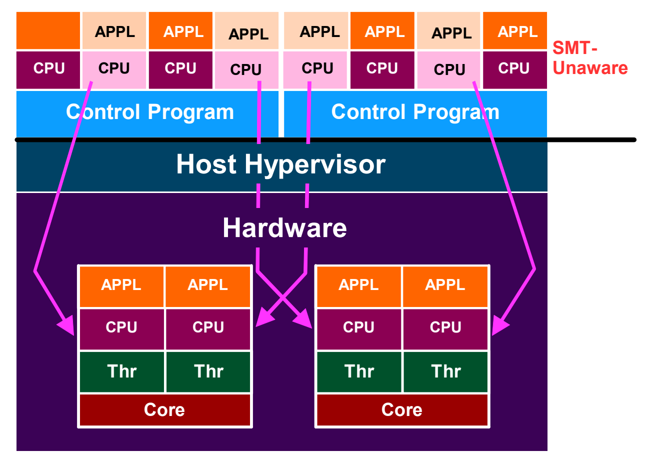

Figure 2 SMT implementation in a cloud infrastructure

This approach to SMT cannot determine runtime capacities, including capacity in use and capacity free. And without the ability to measure runtime capacity in use, accounting and charging for resource consumption becomes disconnected from actual capacity in use, which obscures job costs. Measuring and charging for these capacities typically involves collecting separate measurements for the workload with SMT enabled and disabled. Such separate measurements are sufficient for a simple, repeatable workload with the same characteristics because the hypervisor can pair together any two work units to get repeatable SMT capacity. However, as a workload becomes more complex and unpredictable, the hypervisor ends up grouping different work units together on the same core. Different pairings yield different capacities because the work has different characteristics, so separate measurements fall short of what is needed for measuring and charging for complex and unpredictable workloads.

Because the hypervisor can pair any two control program CPUs together on a core, one control program CPU will observe capacity variability that depends on the characteristics of the other CPU on the same core. A control program has no way to measure, predict, or manage the variable capacity that SMT provides. Yet despite variable capacity, control programs are expected to provide sufficient access to CPU capacity to ensure that applications deliver consistent and predictable response times.

SMT revolution on z Systems

The new IBM® z13™, the latest addition to the IBM z Systems™ family, revolutionizes the design and implementation of SMT through the hardware/software stack to better serve the needs of cloud computing,

The z13 platform supports SMT2 (two threads per core) for a control program that is SMT aware. When a control program enables SMT on the z13, it is certifying that it is SMT aware and will define and manage logical cores and logical threads. As shown in Figure 3, from a hardware perspective, at any moment in time, a single control program manages an entire core and so controls all of that core’s threads. This aspect of the design limits the effects of SMT variability to an individual workload within a control program. The z13 also supports new instrumentation that enables a control program to deliver real-time SMT measurements that can be used for capacity planning and chargeback purposes.

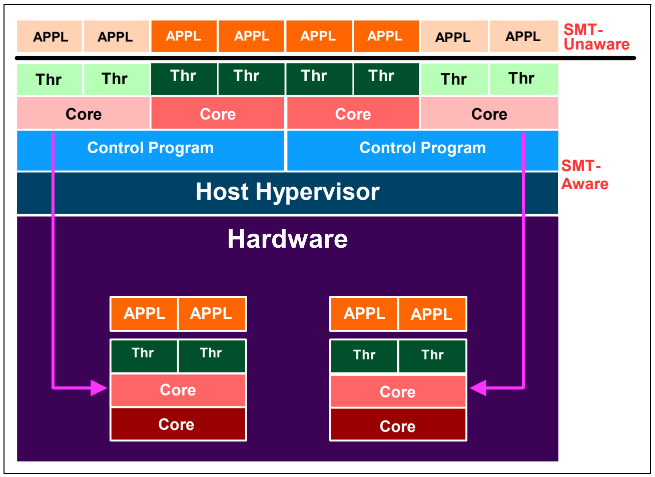

Figure 3 Implementation of SMT on IBM z13

SMT in z/OS on z Systems

On z Systems, a control program manages the whole core, which enables the z Systems operating system, IBM z/OS®, to provide controls to dynamically change the SMT mode (the number of threads per core). For compute-intensive batch workloads, automated processes can switch to SMT Mode 1 (one thread per core) to maximize thread speed. For data-oriented workloads, automated processes can switch to SMT Mode 2 (two threads per core) to maximize core throughput.

z/OS continues to deliver high virtualization with SMT Mode 2 by implementing intelligent expansion and contraction algorithms to maximize core throughput. z/OS also uses the fewest number of cores necessary to meet its application goals, which maximizes available cores for other images.

z/OS also uses the new z13 instrumentation to report:

- SMT Mode 2 core capacity when two threads are in use
- SMT Mode 2 core capacity when one thread is in use (which is identical to SMT Mode 1 core capacity)
- SMT Mode 2 core capacity free when one thread is in use

This information (see Figure 4) allows z/OS to calculate the SMT capacity gain between SMT Mode 2 and SMT Mode 1 at run time for any workload.

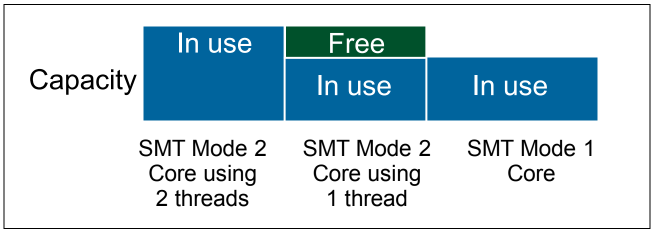
Capacity
In use
Free
In use
In use
SMT Mode 2 Core using 2 threads
SMT Mode 2 Core using 1 thread
SMT Mode 1 Core

Figure 4 z/OS SMT capacity components

z/OS administrators expect a job to require a similar amount of capacity each time that it is run and that it therefore should incur a similar chargeback, regardless of the SMT Mode. With the z13 platform's new instrumentation support, z/OS charges each job according to the amount of single-thread (SMT Mode 1) equivalent capacity it uses, providing consistent chargeback. This design extends throughout z/OS such that all charge back metrics are presented in terms of single-thread (SMT Mode 1) equivalent capacity.

With the z13 and z/OS, the capacity gained from using SMT is just as consumable as any other traditional core capacity. Together, the z13 and z/OS provide the ability to control, monitor, and manage a workload while efficiently using the additional SMT capacity to maximize the number of applications that meet their goals.

Getting started with z/OS SMT

You must keep several considerations in mind when implementing SMT for z/OS.

Installing z/OS SMT support

z/OS support for SMT is available for System z Integrated Information Processors (zIIPs) on z/OS V2R1 and requires installing PTFs for APARS OA43622 and OA43366.

zIIP eligible workloads include but are not limited to Java, XML System Services, z/OS System Data Mover, Common Information Model (CIM), and IBM DB2® Distributed Relational Database Architecture™ (DRDA®) over TCP/IP. For a complete list of workloads and products that run on zIIPs, see the Authorized Use Table URL in the Resources section.

Planning for SMT with z/OS

Before activating SMT Mode 2 on z/OS, you must install the latest versions of your performance monitoring products for SMT Mode 2 capacity and chargeback support.

Determining SMT Benefits with z/OS

A workload benefits from SMT Mode 2 whenever the following occurs:

- The zIIP capacity increases compared to SMT Mode 1
- The slower zIIP thread execution speed results in equal (and sometimes better) response time compared to SMT Mode 1

Generally, use SMT Mode 2 for any workload that can benefit from it. See the documentation for your performance monitoring products for more information about how to assess the SMT benefits for a workload under z/OS.

SMT in z/VM on z Systems

By using SMT, IBM z/VM® (the z Systems hypervisor) can optimize core resources for increased capacity and throughput. Its exploitation of SMT enables z/VM to dispatch a guest (virtual) CPU or z/VM Control Program task on an individual thread (CPU) of an Integrated Facility for Linux (IFL) processor core. This allows the core to be shared by multiple guest CPUs or z/VM Control Program tasks (see Figure 5). For a complete list of workloads and products that run on IFLs, see the Authorized Use Table URL in the Resources section.

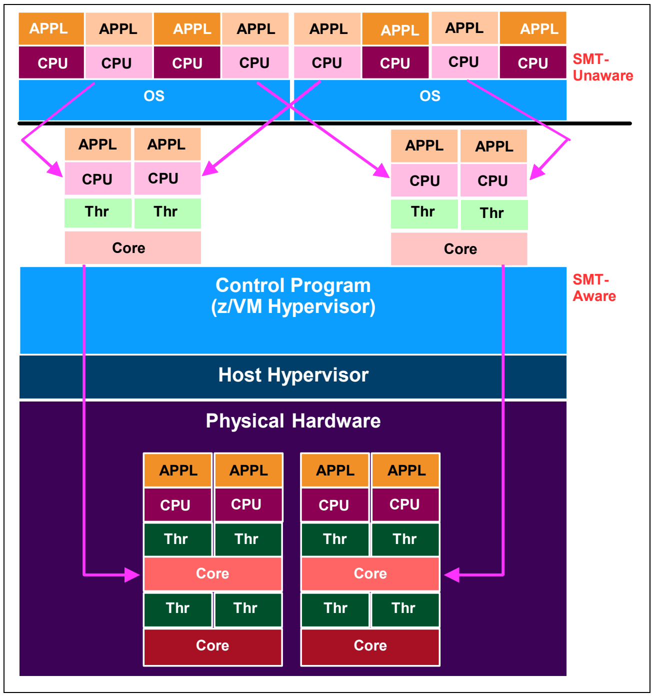

Figure 5 Implementation of z/VM SMT on IBM z13

Getting started with z/VM SMT

Be sure to consider these factors when implementing SMT for z/VM.

Installing z/VM SMT support

Using SMT with z/VM requires that z/VM V6R3 be installed and that the z/VM SMT support PTFs for APARs VM65577 and VM65586 are applied. These support PTFs will be available soon after general availability of the z13 platform.

Planning for SMT with z/VM

Some minor operational procedures will sometimes need to be changed when SMT is enabled for z/VM. In addition, to ensure that guests continue to operate as expected, their configurations must be evaluated and might need to be adjusted. The same applies to CPU pool configurations.

Here are the details:

- Operational considerations

  In a z/VM Single System Image (SSI) environment, systems with SMT enabled can be clustered with those that are not SMT-enabled. Guests can be relocated between SMT and non-SMT systems in the cluster by using Live Guest Relocation (LGR).

- Guest configuration considerations

  As mentioned earlier, threads sharing a core tend to run more slowly, which can affect virtual machine throughput. For example, a virtual machine with two virtual CPUs that consume 100% of two cores with SMT disabled might only be able to consume 70% of two cores with SMT enabled. To consume its pre-SMT capacity, the virtual machine would need to have a third virtual CPU configured. Of course, for this to work, the application that is running in the virtual machine would have to be capable of using the resources of an additional CPU effectively.

  Performance reports that show average and peak CPU utilizations can be used to identify virtual machines that can benefit from additional virtual CPUs in their configurations. Another option is to look for increased CPU delays in virtual machines when SMT is enabled.

  Regardless of whether SMT is involved, when LGR is used in an SSI environment, ensure that the guest configuration is appropriate for the two systems that are involved in the relocation. This might require dynamically adjusting the number of guest virtual CPUs that are in use, either before or after relocation occurs.

- CPU pool configuration considerations

  CPU pools provide a mechanism for limiting CPU resource use by a group of virtual machines to a specific amount. Without SMT, such an expression of the limit is interpreted as a number of cores. With SMT, it is interpreted as a number of threads. IBM intends to change this interpretation to treat pool limits as cores, independent of SMT, but this function will not be available when z/VM SMT support is delivered. So until that occurs, using SMT might require that the CPU pool limits be increased temporarily, in much the same way that it might be necessary to increase a guest’s number of virtual CPUs.

Determining SMT benefits with z/VM

For z/VM, SMT provides more CPUs on which more work can be performed in parallel, meaning increased overall throughput with the z13 compared to the same configuration without SMT enabled. However, individual virtual machine throughput might decrease unless additional virtual CPUs are configured and can be used effectively by the applications they run. Like z/OS, z/VM reports the SMT capacity at run time for any workload (Figure 4 on page 3).

Describing the methods for a comprehensive assessment of SMT benefits in a particular z/VM environment is beyond the scope of this document. The *z/VM Performance Report* (see Resources section) provides more details about how to make such an assessment.

What’s next: How IBM can help

Simultaneous multithreading a feature of the new IBM z13 platform, which has been designed to provide the infrastructure foundation to support demanding workloads, including cloud, alongside your traditional mission-critical applications. Cloud applications hunger for capacity and the z13 delivers more of it by becoming the first system in the z Systems family to support SMT to achieve workload-dependent capacity gains.

The new z13, when combined with z/OS V2R1 and z/VM V6R3, provides a cloud-ready, enterprise class, intelligent, and revolutionary SMT solution. Together, these products deliver the world's first SMT runtime capacity planning metrics for any workload. And its implementation limits the effects of SMT variability to an individual workload within a z/OS or z/VM image, and makes the workload-dependent capacity gains from SMT just as comprehensible and consumable as traditional capacity.

If you are running z/OS or z/VM today, the z13 platform's implementation of SMT has potential to bring even greater efficiency to your organization. IBM has broad experience in optimizing workload throughput and can help you implement SMT in various ways:

- Providing education and training in the use of SMT
- Assessing the benefits of using SMT on the z13 using z/VM, z/OS, or both
- Implementing SMT in your cloud infrastructure

For more information about using the z13 and SMT, consult your local IBM representative.

Resources for more information

For customers considering the IBM z13:

- *Why System z® might be exactly what your business needs* (newsletter)

  http://www.ibm.com/vrm/newsletter_11421_9000334_241332_email_DYN_1IN/DIbm201466859

- IBM zEnterprise® System

  http://www-03.ibm.com/systems/z/hardware/zenterprise/

For customers exploring z Systems or z/OS:

- *Back to the future: Why z/OS mainframe is the ideal cloud platform* (blog post)

  http://thoughtsoncloud.com/2014/05/back-future-zos-mainframe-ideal-cloud-platform/

- *Cloud Workloads On The Mainframe* (IBM Redbooks® Point of View)

  http://www.redbooks.ibm.com/abstracts/redp5108.html?Open

Additional z/OS resources:

- z/OS MVS™ Knowledge Center:

  http://pic.dhe.ibm.com/infocenter/zos/v2r1/topic/com.ibm.zos.v2r1.iea/iea.htm

- z/OS Introduction and Release Guide:

  http://www-01.ibm.com/support/knowledgecenter/SSLTBW_2.1.0/com.ibm.zos.v2r1.e0za100/toc.htm?lang=en

- Resource Measurement Facility™ Report Analysis (for z/OS installations with RMF™):

  http://www-01.ibm.com/support/knowledgecenter/SSLTBW_2.1.0/com.ibm.zos.v2r1.erbb500/erb2ra00.htm

Additional z/VM resources:

- z/VM Publications Library

  http://www.vm.ibm.com/library/zvmpdf.html

- z/VM V6R3 Library in IBM Knowledge Center

  http://www-01.ibm.com/support/knowledgecenter/SSB27U_6.3.0/com.ibm.zvm.v630/zvminfoc03.htm

- z/VM Performance Report

  http://www.vm.ibm.com/perf/reports/zvm/html/

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-5144-00, 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®
Distributed Relational Database Architecture™
DRDA®
IBM®
MVS™
Redbooks®
Redbooks (logo)
Resource Measurement Facility™
RMF™
System z®
z/OS®
z/VM®
zEnterprise®

The following terms are trademarks of other companies:

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