

CICS Scalability

Catherine Moxey
IBM

Tuesday, February 5, 2013
Session Number 12433



Disclaimers

© IBM Corporation 2013. All Rights Reserved.



IBM's statements regarding its plans, directions, and intent are subject to change or withdrawal at IBM's sole discretion. Information regarding potential future products is intended to outline our general product direction and it should not be relied on in making a purchasing decision. Any information mentioned regarding potential future products is not a commitment, promise, or legal obligation to deliver any material, code or functionality. Information about potential future products may not be incorporated into any contract. The development, release, and timing of any future features or functionality described for our products remains at our sole discretion.

The session and materials has been prepared by IBM or the session speaker and reflect their own views. They are provided for informational purposes only, and are neither intended to, nor shall have the effect of being, legal or other guidance or advice to any participant. While efforts were made to verify the completeness and accuracy of the information contained in this presentation, it is provided AS IS without warranty of any kind, express or implied. IBM shall not be responsible for any damages arising out of the use of, or otherwise related to, this presentation or any other materials. Nothing contained in this presentation is intended to, nor shall have the effect of, creating any warranties or representations from IBM or its suppliers or licensors, or altering the terms and conditions of the applicable license agreement governing the use of IBM software.

References in this presentation to IBM products, programs, or services do not imply that they will be available in all countries in which IBM operates. Product release dates and/or capabilities referenced in this presentation may change at any time at IBM's sole discretion based on market opportunities or other factors, and are not intended to be a commitment to future product or feature availability in any way. Nothing contained in these materials is intended to, nor shall have the effect of, stating or implying that any activities undertaken by you will result in any specific sales, revenue growth or other results.

Performance is based on measurements and projections using standard IBM benchmarks in a controlled environment. The actual throughput or performance that any user will experience will vary depending upon many factors, including considerations such as the amount of multiprogramming in the user's job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve results similar to those stated here. All customer examples described are presented as illustrations of how those customers have used IBM products and the results they may have achieved. Actual environmental costs and performance characteristics may vary by customer.

Complete your sessions evaluation online at SHARE.org/SFEval





Trademarks

IBM, the IBM logo, ibm.com, AppScan, CICS, Cloudburst, Cognos, CPLEX, DataPower, DB2, FileNet, ILOG, IMS, InfoSphere, Lotus, Lotus Notes, Maximo, Quickr, Rational, Rational Team Concert, Sametime, Tivoli, WebSphere, and z/OS are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both. If these and other IBM trademarked terms are marked on their first occurrence in this information with a trademark symbol (® or ™), these symbols indicate U.S. registered or common law trademarks owned by IBM at the time this information was published. Such trademarks may also be registered or common law trademarks in other countries.

A current list of IBM trademarks is available on the Web at “Copyright and trademark information” at ibm.com/legal/copytrade.shtml.

Coremetrics is a trademark or registered trademark of Coremetrics, Inc., an IBM Company.

SPSS is a trademark or registered trademark of SPSS, Inc. (or its affiliates), an IBM Company.

Unica is a trademark or registered trademark of Unica Corporation, an IBM Company.

Java and all Java-based trademarks and logos are trademarks of Oracle and/or its affiliates. Other company, product and service names may be trademarks or service marks of others. References in this publication to IBM products and services do not imply that IBM intends to make them available in all countries in which IBM operates.

Session Abstract

- Increased operational efficiency is a fundamental component of CICS TS V5.1. Significant improvements to the horizontal and vertical scalability of CICS are delivered through a number of enhancements, ranging from improved support for OTE (threadsafe), greater use of 64-bit storage, optimized TCB usage, and changes to a region's MAXTASK – to name a few. This session will provide a summary of the improvements in CICS version 4, as well as a detailed look at the enhancements and benefits available in CICS TS V5.1.

Agenda

- Operational Efficiency Overview
 - CICS TS V4 and V5.1
- CICS Scalability items
 - Simplified configuration and increased max. task limit
 - Enhanced Instrumentation
 - Java 7 support
 - OTE and threadsafety
 - VSCR and 64-bit exploitation
 - Access to 64-bit storage from assembler programs
- Scalability benefits from z/OS
- Summary

CICS V5.1

Operational Efficiency

- *Greater capacity* - achieve cost savings through consolidation
- *Managed operations* - control critical resource thresholds with policies
- *Increased availability* - reduce the need for planned downtime
- *Deeper insight* - extend performance and compliance information

100+
requirements
satisfied!

Service Agility

- *First-class applications* - create agile services from existing assets
- *First-class platforms* - create agile service delivery platforms
- *Modern interfaces* - build rich web experiences for critical applications
- *Foundational enhancements* - extend core capabilities

... with Cloud Enablement

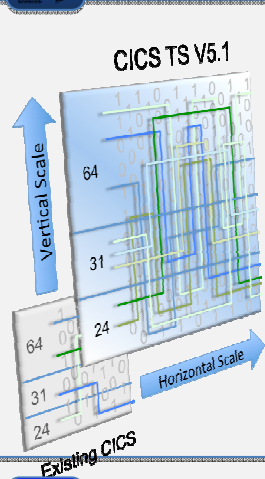
Consistent with the IBM Cloud Computing strategy
Positioning customers for the next transformational era in technology
Moving towards a cloud oriented service delivery platform



CICS TS V5.1 : Driving Operational Efficiency



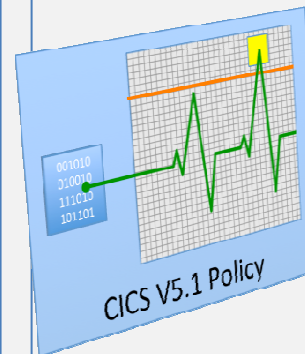
Greater Capacity



- Doubling of the MAXTASK limit to 2,000
- Increased 64-bit and reduced 24-bit storage usage
- Greater parallelism from threadsafe API and SPI
- Greater system parallelism through optimized TCB usage
- Performance improvements from 64-bit Java 7
- Greater access to 64-bit storage from Assembler programs



Managed Operations



- Automated control over critical system resources
- Set data access thresholds on SQL or file access
- Set program loop thresholds on EXEC LINK
- Set storage request thresholds
- Set CPU time thresholds
- Policies can issue messages, abending tasks, or create events



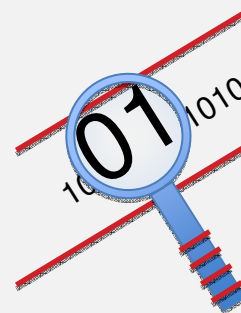
Increased Availability



- Upgrade CICS versions and releases without requiring a z/OS restart
- Refresh Secure Sockets Layer (SSL) certificates
- Keep IPIC connections up and running
- Support more IBM GDPS/AA solutions
- Dynamically specify cross-system coupling facility groups
- Better reflect current best practices with updated and simplified defaults



Deeper Insight



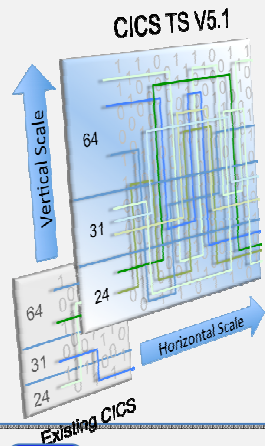
- Auditing of SPI commands that alter the system
- Improved auditing of user IDs that make requests over IP
- Extended identity propagation to include started tasks
- Cipher suites used for SSL connections stored in the performance records
- Calculation of the actual and potential use of specialty processors
- Regular status updates provided while lost locks recovery is taking place



CICS TS 4 : Driving Operational Efficiency



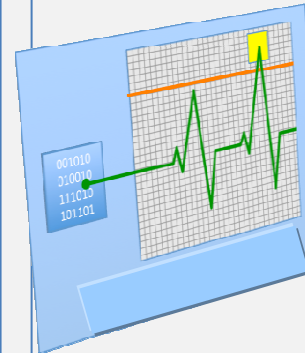
Greater Capacity



- Multi-processor exploitation
 - File control and TS requests threadsafe when function shipped over IPIC
 - CONCURRENCY(REQUIRED) option to minimize TCB switching
 - Threadsafe CICS-DBCTL with IMS 12
- Internal trace table and TS Main in above-the-bar storage, and other 64-bit exploitation
- Parsing of web services requests optimized with use of XML system services parser – zAAP offloadable



Managed Operations



- Transaction Tracking to track and correlate the progress of tasks across multiple CICS regions
- Sysplex-optimized workload management
- New location-neutral workload management algorithms
- Discovery Library Adapter to extract CICS configuration information used in Tivoli tooling



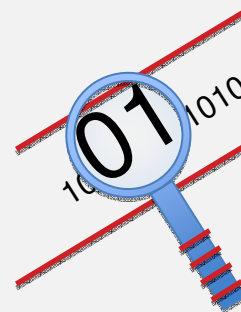
Increased Availability



- HTTP inbound connection throttling
- HTTP outbound connection pooling
- Support for attaching to a WebSphere MQ queue sharing group rather than a specific queue manager, and for Group Unit of Work recovery



Deeper Insight

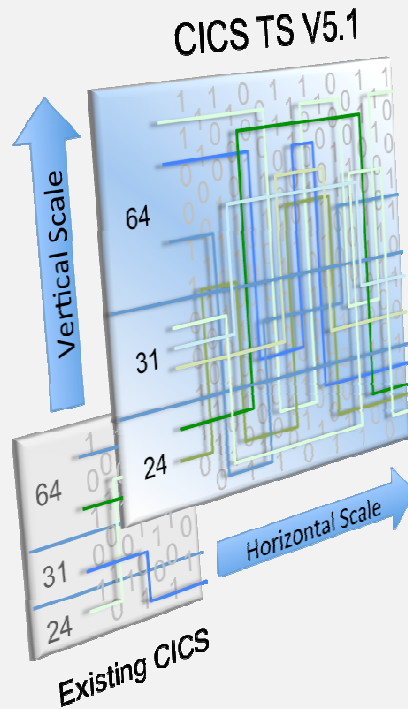


- Resource Signatures to audit when and how resources were last defined / installed
- z/OS Identity Propagation support, to propagate distributed identities for audit etc.
- Support for up to 100-character password phrases



Greater capacity

Achieve cost savings through consolidation



- Simplified configuration and increased max. task limit
- Enhanced Instrumentation
- Java 7 support
- OTE and threadsafety
- VSCR and 64-bit exploitation
- Access to 64-bit storage from assembler programs

Driving operational efficiencies – Greater capacity

Run more, more easily

V
E
R
T
I
C
A
L

S
C
A
L
I
N
G

- Vertical Scaling
 - Relieve region storage constraints
 - Further virtual storage constraint relief
 - Maximum task limit doubled
 - Further threadsafe support to reduce TCB switching and increase workload capacity
- Horizontal Scaling
 - Instrumentation enhancements – understand how the platform is scaling
 - Standardization and simplification



'right-size' and simplify CICS topologies

HORIZONTAL SCALING



Notes ...

Greater capacity is delivered through significant vertical and horizontal scalability enhancements. Fewer regions can now run the same workload. They provide the opportunity for cost savings through consolidation. Reducing the number of regions can deliver savings in both CPU consumption and administration overheads.



Greater Capacity – Achieve cost savings through consolidation

IBM Lab benchmark demonstrated...

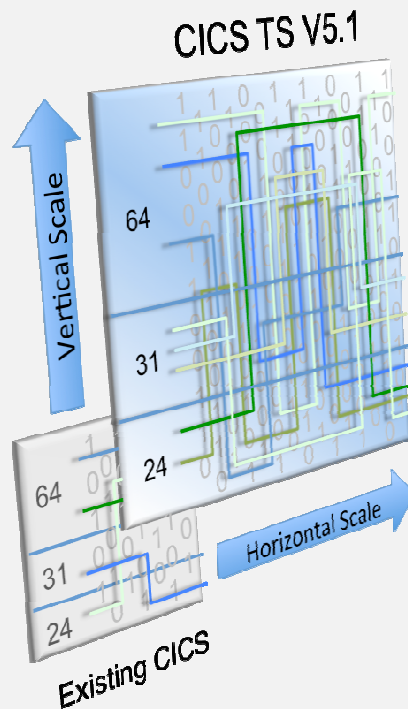
- **Consolidate 30 regions down to 10***
- **Decrease CPU usage by 10%***
- **Reduce the management burden by 2/3***
- **Maintain the same workload***



*Test conducted under lab conditions – For further information contact IBM



Greater capacity



- Simplified configuration and increased max. task limit
- Enhanced Instrumentation
- Java 7 support
- OTE and threadsafety
- VSCR and 64-bit exploitation
- Access to 64-bit storage from assembler programs



Simplified Configuration

- CICS TS V5.1 introduces a number of enhancements to make operations easier and more transparent
- For example
 - the default values for several CICS startup parameters are changed to adopt best practice
 - or are removed altogether where CICS is now able to automatically make adjustments at run time



Notes ...

Some CICS® configuration and default values are changed to provide best operational practices and to reduce the amount of manual configuration required by system programmers.



Increased max. tasks limit

- MAXTASK
 - Single region capacity being constrained by CICS maxtask limit
 - Maximum tasks limit increased from 999 to 2000 in CICS TS V5.1
 - Default value changed from 5 to 500
 - Minimum increased from 1 to 10
 - Primarily for Terminal and File Owning Regions (TORs and FORs)
 - Single TOR routing to multiple Application Owning Regions (AORs)
 - Single FOR servicing multiple Application Owning Regions (AORs)
 - Value now used to set some of the MAXxxxTCBS parameters



Notes ...

The maximum number of user tasks that can exist in a CICS® region at the same time is increased from 999 to 2000. The minimum value is increased from 1 to 10, and the default value is increased from 5 to 500. The changes mean that a CICS region operates more efficiently with the default setting and can process more workload, so the need to increase the number of CICS regions is reduced.

These changes apply to the MXT system initialization parameter, the MAXTASKS option of the SET SYSTEM and CEMT SET SYSTEM commands, and the MAXTASKS value in CICSplex® SM.

You must ensure that enough storage is available to support the maximum number of tasks value. When you increase the maximum number of tasks for a CICS region, measure performance to ensure that the response time and other time components (such as dispatch time and suspend time) for your transactions remain acceptable. In some systems, an increase in concurrent tasks might increase resource contention to a level that causes additional delays for transactions. In the performance class data for a transaction, the new MAXTASKS field records the current setting for the maximum number of tasks for the CICS region, and the CURTASKS field records the current number of active user transactions in the system at the time the user task was attached. This data helps you to assess the relationship between the task load during the life of a transaction, and the performance of the transaction.



System Parameter Simplification

- MAXxxTCBs Simplification
 - MAXOPENTCBS, MAXXPTCBS, MAXJVMTCBS
 - System Initialization Parameters now obsolete
 - MAXOPENTCBS
 - *Set internally to a value of $(2 * MXT \text{ Value}) + 32$*
 - MAXXPTCBS
 - *Set internally to the MXT Value*
 - MAXJVMTCBS
 - *No longer applies due to removal of support for pooled JVMs*
- CEMT and EXEC CICS INQ/SET DISPATCHER
 - MAXOPENTCBS and MAXXPTCBS settings
 - *Available on INQUIRE DISPATCHER only*
 - *SPI SET will return RESP(0) but do nothing*

Notes ...

Before CICS TS for z/OS, Version 5.1, you used the MAXOPENTCBS and MAXXPTCBS system initialization parameters to set limits for the number of open TCBs in the CICS region. MAXOPENTCBS specified the maximum number of open TCBs in the pool of L8 and L9 mode TCBs, which are used for OPENAPI application programs and task-related user exits that are enabled with the OPENAPI option. MAXXPTCBS specified the maximum number of open TCBs in the pool of X8 and X9 mode TCBs, which are used for C and C++ programs compiled with the XPLINK option.

CICS now automatically sets and changes the limits for open TCBs in these TCB pools when you set or change the maximum number of tasks (MXT or MAXTASKS) specified for the CICS region. For the limit on the L8 and L9 mode open TCB pool, CICS uses the following formula: $(2 * \text{MXT Value}) + 32$ For the limit on the XP TCB pool, CICS uses a value equal to the MXT value for the CICS region.



System Parameter Modernization

- ICVTSD – Terminal Scan Delay
 - Default changed from 500ms to 0ms
- Priority Aging – PRTYAGE
 - Default changed from 32768ms to 1000ms
- AKPFREQ – Activity Keypoint Frequency
 - Minimum lowered from 200 to 50
- DSALIM
 - Default value unchanged at 5MB
- EDSALIM
 - Default value changed from 48MB to 800MB (megabytes)
- MEMLIMIT (specified in JCL)
 - Minimum required is now 6 GB

Notes ...

Before CICS TS for z/OS, Version 5.1, the default value for the ICVTSD system initialization parameter, which specifies the terminal scan delay time, was 500 milliseconds. In CICS TS for z/OS, Version 5.1, the default value is 0. The terminal scan delay facility was used in earlier releases to limit how quickly CICS dealt with some types of terminal output requests made by applications, in order to spread the overhead for dealing with the requests. Specifying a nonzero value was sometimes appropriate where the CICS system used non-SNA networks. However, with SNA and IPIC networks, setting ICVTSD to 0 is appropriate to provide a better response time and best virtual storage usage.

The default value of the priority aging factor, which is set by using the PRTYAGE system initialization parameter, is reduced from 32768 milliseconds (32.786 seconds) to 1000 milliseconds (1 second). Therefore, by default, the priority of long-running tasks that are on the ready queue will increase more rapidly.

The minimum value in the activity keypoint frequency range is reduced from 200 to 50. The activity keypoint frequency is set by using AKPFREQ system initialization parameter. The reduced minimum value means that completed log task records can be deleted more frequently, which reduces the DASD dataspace usage. Therefore, the primary system logstream remains at a reasonable size.

The EDSALIM system initialization parameter specifies the upper limit of the total amount of storage within which CICS can allocate the individual extended dynamic storage areas (EDSAs) that reside in 31-bit (above-the-line) storage. Before CICS TS for z/OS®, Version 5.1, the default value for this system initialization parameter was set at the minimum required to start a CICS region, which is 48 MB. The default value is now increased to 800 MB, which enables a CICS region started with the default value to process a reasonable workload. You can tune the EDSALIM value in a running CICS system.



Simplified Configuration

- STATINT – Statistics Collection Interval
 - Default changed from 030000 to 010000 (hhmmss)
- TCTUALOC – Terminal User Area Location
 - TCT User Area Location
 - Default changed from TCTUALOC=BELOW to TCTUALOC=ANY
- TRTRANSZ – Transaction Dump Trace Table Size
 - Default increased from 16K to 1024K (1MB)
 - Trace table is in 64-bit storage
 - Giving much better chance of tracing the error before a dump



Notes ...

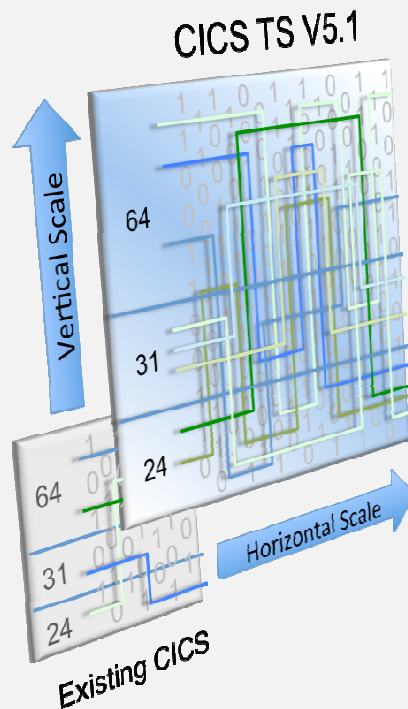
To help you collect valuable CICS system and resource data in a timely manner, the default stats interval has been reduced to 1hr.

The TCTUALLOC system initialization parameter specifies where terminal user areas (TCTUA) are to be stored. Before CICS TS for z/OS, Version 5.1, the default setting specified that terminal user areas were always stored in 24-bit (below-the-line) storage. The default setting is now ANY, which means that CICS uses 31-bit (above-the-line) storage to store terminal user areas whenever possible, and 24-bit (below-the-line) storage as an alternative.

Before CICS TS for z/OS, Version 5.1, the default value of the TRTRANSZ parameter was 16 KB. The default value is now increased to 1024 KB. This value provides a larger transaction dump trace table, which might contain more useful trace information. Check your current setting for the z/OS parameter MEMLIMIT, which limits the amount of 64-bit storage that the CICS address space can use. Your setting for TRTRANSZ must remain within MEMLIMIT, and you must also allow for other use of 64-bit storage in the CICS region.



Greater capacity



- Simplified configuration and increased max. task limit
- **Enhanced Instrumentation**
- Java 7 support
- OTE and threadsafety
- VSCR and 64-bit exploitation
- Access to 64-bit storage from assembler programs

Instrumentation



- CICS statistics and monitoring provide vital information about the health of the system and the workloads.
- CICS TS V5.1 statistics now include more data about the load, capacity, and performance of the system.
- The data provided by monitoring can help you to assess performance more accurately, including potential bottlenecks.
- Together these enhancements enable you to make more informed decisions about hardware and software upgrades, and application deployments.



Notes ...

CICS statistics are enhanced to provide more information that relates to the load, capacity, and performance of the system. This information can aid decisions about vertical and horizontal scalability.

CICS monitoring provides additional performance data and system information to help you to assess the performance of your CICS regions accurately, identify potential bottlenecks, and make informed decisions about hardware and software upgrades and application deployment.

Instrumentation Enhancements – Monitoring



- Physical hardware environment
 - CEC Machine Type and Model ID
 - e.g. 2097-740
- Transaction performance related to CICS region load
 - Current active task count and maxtask setting
- Improved transaction wait (suspend) analysis
 - MRO/ISC Allocate Waits
 - IPIC Allocate Waits
 - RO TCB and SO TCB Mode Delays
 - Intrapartition and Extrapartition TD Lock Waits
 - File Control Exclusive Control Waits
 - VSAM File String Waits

Notes ...

The new CECMCHTP and CECMDLID fields in the DFHTASK performance class group for a transaction show the CEC machine type and CEC model number for the physical hardware environment where the CICS region is running. CEC (central electronics complex) is a commonly used synonym for CPC (central processing complex), which refers to a collection of physical hardware including main storage, one or more central processors, timers, and channels.

You can use this information with the IBM® Large Systems Performance Reference (LSPR) ratios to make an accurate assessment of CICS performance and relative processor capacity, particularly when considering upgrades to your z/OS® hardware.

The new MAXTASKS field in the DFHTASK performance class group for a transaction records the current setting for the maximum number of tasks for the CICS region. The CURTASKS field records the current number of active user transactions in the system at the time the user task was attached. This data helps with assessing the relationship between the task load during the life of a transaction, and the performance of the transaction.

New Performance Class monitoring fields have been added to provide information on the time spent by user tasks in various waits:

- MRO, LU6.1, and LU6.2 session allocation waits
- IPIC session allocation waits
- RO TCB mode delays
- SO TCB mode delays
- Transient data intrapartition lock waits
- Transient data extrapartition lock waits
- File control waits for exclusive control of a VSAM control interval
- File control waits for a VSAM string



Instrumentation Enhancements – Offload times

- zAAP/zIIP Specialty Processor Transaction CPU time
 - Existing CMF Performance Class Field
 - a) “USRCPUT” → Total CPU time used on a standard CP, System z Application Assist Processor (zAAP), or System z Integrated Information Processor (zIIP)
 - New CMF Performance Class Fields
 - b) “CPUTONCP” → Total CPU time on standard CP
 - c) “OFFLCPUT” → Total Offload CPU time on standard CP (Offload eligible but ran on standard CP)
 - From the new metrics the following can also be derived
 - d) Total CPU time on zAAP/zIIP = (USRCPUT – CPUTONCP)
 - e) Total CPU time on CP that was not offload = (CPUTONCP – OFFLCPUT)
 - f) Total CPU time offload eligible = (OFFLCPUT + d)
 - Requires
 - z/OS R13 APAR OA38409 and IBM System z9 or later

Notes ...

The new CPUTONCP and OFFLCPUT fields in the DFHTASK performance class group for a transaction can be used to calculate the processor time that a task spends on a zIIP or zAAP specialty processor, and also show you the processor time that the task could have spent on a specialty processor.

- Field 436, CPUTONCP, shows the total task processor time spent on a standard processor. To calculate the task processor time spent on a specialty processor, subtract the time recorded in this field from the time recorded in field 008, USRCPUT.
- Field 437, OFFLCPUT, shows the total task processor time that was eligible for offload to a specialty processor, but actually ran on a standard processor. To calculate the total task processor time that was not eligible for offload, subtract the time recorded in this field from the time recorded in field 436, CPUTONCP.
- To calculate the total task processor time that was either actually spent on a specialty processor, or eligible to be spent on a specialty processor, use the following equation: $(OFFLCPUT + (USRCPUT - CPUTONCP))$

Note: The times shown in the CPUTONCP and OFFLCPUT fields are only available when running on a system that supports the Extract CPU Time instruction service that is available on IBM System z9® or later hardware. For z/OS, Version 1 Release 13, the PTF for APAR OA38409 must also be applied.

Instrumentation Enhancements – Monitoring of Application and Policy information; SSL and RMI



- Application Context
 - Platform name
 - Application name
 - Operation name
 - Major, Minor, and Micro version numbers
- Policy
 - Policy threshold exceeded count
- SSL CIPHER code
 - SSL ciphers used are now recorded in SMF 110 CMF performance class records for better performance analysis
- Monitoring RMI Data Collection Option
 - Additional performance metrics on CICS Resource Manager usage
 - Default changed from RMI=NO to RMI=YES

Notes ...

Further monitoring enhancements include:

New monitoring fields to provide application context data to monitor applications in CICS, providing details about the platform, the application and its version, and the operation.

A new Performance class data field in DFHCICS provides the number of policy thresholds that the task has exceeded.

A single cipher is negotiated during an SSL handshake between a server and a client. That cipher is used for all subsequent traffic on the SSL connection. CICS now displays the code for the cipher suite in the performance data field SOCIPHER in the DFH SOCK group. You can use this information to identify any cipher suites that are offered by the CICS region but are not being selected for SSL connections. You can also identify any less efficient or less secure cipher suites that are being selected for SSL connections but that you would prefer to eliminate.

The default value for the MCT Resource Manager Interface (RMI) parameter has changed from RMI=NO to RMI=YES. With the new setting, additional monitoring performance data is collected by default for the resource managers used by your transactions.



Instrumentation Enhancements – 64-bit storage monitoring

- 64-bit storage usage
 - Performance Class data
 - GETMAIN requests and HWM storage for user storage in GCDSA and GUDSA
 - GETMAIN requests, and shared storage obtained and released for shared storage in GCDSA and GSDSA
 - Exception Class data enhancements
 - Storage Waits in GCDSA, GUDSA, and GSDSA



Notes ...

New monitoring fields in the DFHSTOR performance class group provide data on the use of 64-bit (above-the-bar) storage by user tasks, in the CICS GDSAs. The following information is provided:

- A GETMAIN request count and high water mark for user storage obtained by a user task in the GCDSA and in the GUDSA.
- A GETMAIN request count, amount of storage obtained, and amount of storage freed, for shared storage obtained by a user task in the GCDSA or GSDSA.

Resource identifiers for waits for storage in the GUDSA and GSDSA are also added to the exception class data.

Instrumentation Enhancements – Statistics



- Storage Manager Statistics
 - New GxDSAs for 64-bit storage
 - DSA statistics
 - Domain Subpool statistics
 - Task Subpool statistics
- Loader Global Statistics
 - New statistics on RO TCB program load requests and load time
- URIMAP Resource Statistics
 - New URIMAP Usage value → JVMSERVER
- Dispatcher Global Statistics
 - TCB Pools and TCB Modes
 - JVM TCB Pool and J8/J9 TCB Modes Obsolete
- Statistics Data Interval Collection Option
 - Default changed from STATINT=030000 to 010000 (hhmmss)
 - Statistics Recording option STATRCD=NO|YES – default unchanged
 - More timely statistics data collection – peak hour analysis

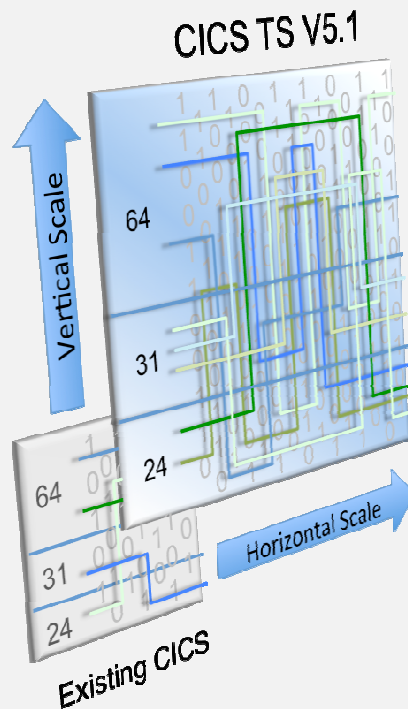


Notes ...

CICS® statistics are enhanced to provide more information that relates to the load, capacity, and performance of the system. This information can aid decisions about vertical and horizontal scalability.



Greater capacity



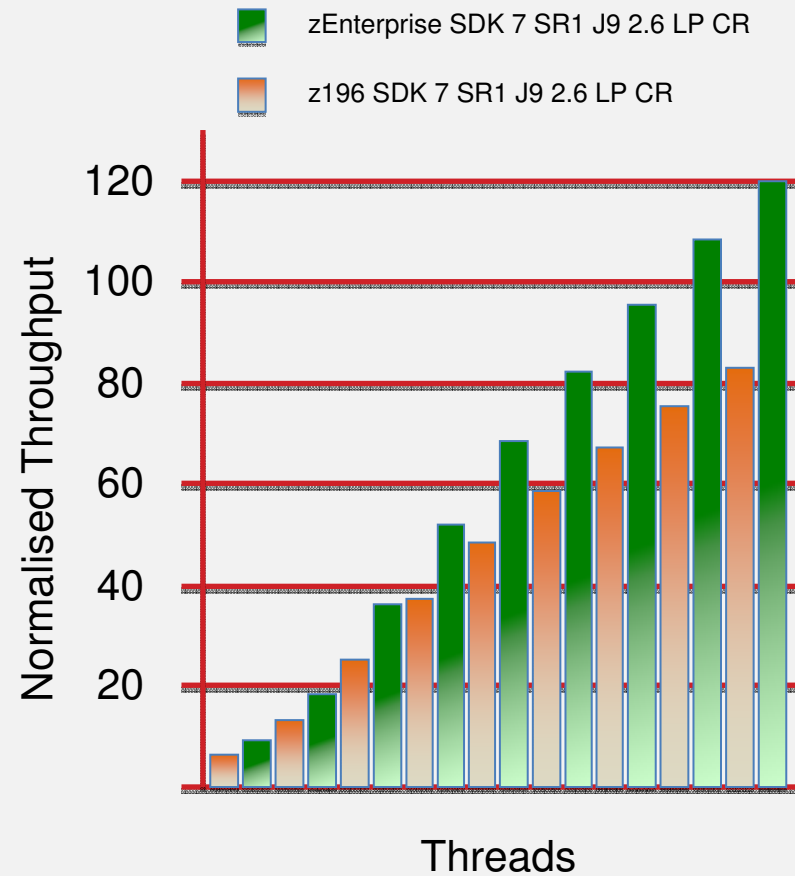
- Simplified configuration and increased max. task limit
- Enhanced Instrumentation
- **Java 7 support**
- OTE and threadsafety
- VSCR and 64-bit exploitation
- Access to 64-bit storage from assembler programs



Performance improvements from Java 7 support

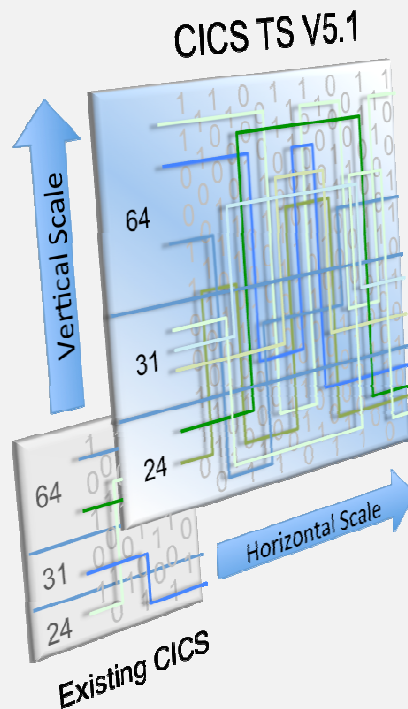
The latest JVM delivers a performance boost...

- zEnterprise EC12 offers a ~45% improvement over z196 running the Java Multi-Threaded Benchmark
- zEC12 has additional instructions specifically for Java





Greater capacity



- Simplified configuration and increased max. task limit
- Enhanced Instrumentation
- Java 7 support
- **OTE and threadsafety**
- VSCR and 64-bit exploitation
- Access to 64-bit storage from assembler programs

Open Transaction Environment – Threadsafe Enhancements



- Threadsafe Transient Data Commands
 - Commands that access Transient Data (TD) are now threadsafe
 - EXEC CICS READQ TD, WRITEQ TD, and DELETEQ TD
- TD Global User Exits must be threadsafe
 - XTDEREQ, XTDEREQC
 - XTDIN, XTDOUT, and XTDREQ
- TD function Shipping over IPIC will use an Open TCB
 - Also drive mirror on open TCB
- System initialization parameter TDSUBTASK obsolete
 - If on QR TCB TD will switch to FO TCB, If on an open TCB it uses the open TCB
- Existing SPI commands commonly used in some applications now threadsafe
 - EXEC CICS SET TASK
 - INQUIRE and SET TRACEDEST / TRACEFLAG / TRACETYPE



Notes ...

The CICS® transient data facility and more API and SPI commands have been made threadsafe, to increase throughput and reduce CPU usage for threadsafe applications that make use of them.

The CICS transient data facility, comprising the WRITEQ TD, READQ TD and DELETEQ TD commands, is now threadsafe, so CICS can process these transient data requests on an open TCB. Transient data requests are also threadsafe when you function ship them to a remote region over an IPICT connection.

To optimize TCB switching and gain the performance benefits of the open transaction environment, global user exit programs that run at the transient data exits XTDEREQ, XTDEREQC, XTDEREQ, XTDIR, and XTDIROUT must be coded to threadsafe standards and defined to CICS as threadsafe.

Because the CICS transient data facility is now threadsafe, the TDSUBTASK system initialization parameter is no longer required. This system initialization parameter specified whether CICS used the FO TCB to write to a particular type of extrapartition transient data queue.

Existing SPI commands made threadsafe in this release

- [INQUIRE TRACEDEST](#)
- [INQUIRE TRACEFLAG](#)
- [INQUIRE TRACETYPE](#)
- [SET TASK](#)
- [SET TRACEDEST](#)
- [SET TRACEFLAG](#)
- [SET TRACETYPE](#)



OTE – Program load and open TCBs

- CICS program LOADs when running on an Open TCB
 - When running on an open TCB and a CICS program load is requested there is no longer a TCB switch to the RO TCB
 - EXEC CICS LINK, LOAD, XCTL, ...
 - Updated Loader global statistics
 - New statistics on RO TCB program load requests and load time
 - Global User Exits must be threadsafe
 - XLDLOAD, XLDELETE, and XRSINDI
 - CICS RO TCB will still be used for
 - CICS program LOADs when NOT running on an Open TCB
 - DFHRPL and LIBRARY Dataset Management
 - Benefits
 - Reduced contention for the single CICS RO TCB
 - Reduced pathlength – RO TCB switch eliminated
 - Significantly increased potential CICS program LOAD capacity

Notes ...

When an application that is currently running on an open TCB issues a command that loads a program, CICS® no longer switches to the RO (resource-owning) TCB to load the program. Instead, CICS carries out the program load on the open TCB.

If an application that is not running on an open TCB loads a program, the RO TCB is still used. The RO TCB is also used if a data set containing programs needs to be opened or closed.

The new process has several benefits that might produce improvements in performance and throughput for applications in your CICS system:

- Reduction in TCB switching
- Increased concurrency for program loading operations
- Increased availability of the RO TCB



OTE – Reductions in TCB switching

- Removed TCB switch for Java applications accessing DB2
 - Java applications that use JDBC or SQLJ will not require a TCB switch to L8
 - Java programs will perform better due to reduced TCB switching
- Reduction in TCB switching requirements
 - The problem – T8/X8 applications switch TCBs to access DB2
 - T8 – Java applications in a JVM Server using JDBC or SQLJ
 - X8 – CICS-key XPLink programs (C/C++)
 - TCB switch to L8 no longer needed to access DB2
 - Supports inter-language program LINKs
 - End of task syncpoint will still use an L8
 - Required PTFs for DB2 V9 (UK78500) and V10 (UK78499)

Notes ...

Applications can now access JDBC and SQLJ from the T8 TCB instead of forcing a move to L8 TCB.

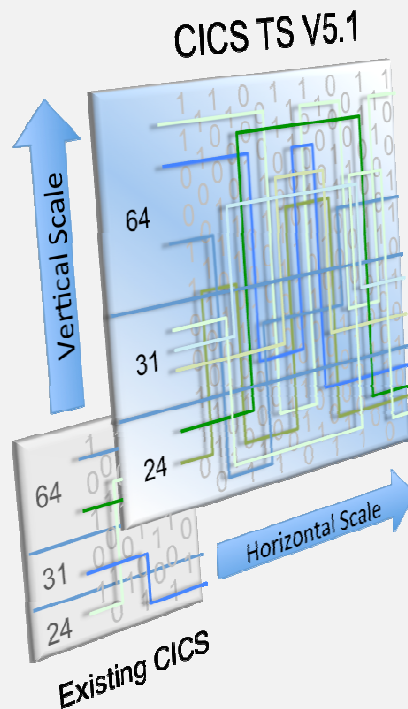
The CICS-DB2 task-related user exit has been changed to take advantage of an enhancement to the RMI that allows TRUEs to run on any key 8 TCB, not just L8 and to be able to move its thread from one TCB to another. With this enhancement, the same thread can be used by Java and non Java application programs in the same transaction.

A new option **REQUIRED** on the **ENABLE PROGRAM** command is used to specify that a task-related user exit must run on an open TCB. If **OPENAPI** is specified with **REQUIRED**, then an L8 TCB is used. If **OPENAPI** is not specified, then any key 8 open TCB is used.

The change in TCB switch behavior affects the results you see in CICS monitoring and statistics. TCB usage for Java DB2 applications is changed in that DB2 CPU time is now accumulated against a T8 TCB. End of task syncpoint processing will still be accumulated on an L8 TCB.



Greater capacity



- Simplified configuration and increased max. task limit
- Enhanced Instrumentation
- Java 7 support
- OTE and threadsafety
- **VSCR and 64-bit exploitation**
- Access to 64-bit storage from assembler programs



24-bit Virtual Storage Constraint Relief (VSCR)

- 24-bit Virtual Storage Constraint Relief
 - Reduce pressure on below the line storage
 - Reduce below the line Short-on-storage conditions
 - Provide for greater capacity for workload growth
- 24-bit Virtual Storage Constraint Relief
 - Control blocks, Modules, and stack storage moved above line
 - Syncpoint, Transient Data, Journal Control, ...
 - Transient Data access method buffers
 - Extrapartition transient data – moved from 24-bit to 31-bit
 - Reduce below-the-line storage used by CICS supplied transactions
 - Redefined with TASKDATALOC(ANY)
 - For example
 - *CEMT, CEOT, CESN, CESF, CETR, CMSG, CRTE, ...*
 - *CWTO, CIEP, CSNC, and the Mirror transactions ...*
 - *CEDF and CECI processing*



Notes ...

Several elements of the CICS® infrastructure now use 31-bit (above-the-line) storage in the CICS region, in place of 24-bit (below-the-line) storage. Lowering the demands on 24-bit storage helps avoid short-on-storage conditions and can reduce the need for additional CICS regions.

The CICS runtime environment is thus improved in the following ways:

- The need for additional CICS regions to resolve demands on 24-bit storage is reduced.
- Problems caused by short-on-storage conditions in 24-bit storage are avoided.
- CICS regions can carry out simultaneous processing of a greater number of tasks that use these CICS infrastructure items.



Further 24-bit VSCR

- Mirror transactions
 - Supplied mirror transaction defined with TASKDATALOC(ANY)
 - Will use 31-bit task storage
 - AEZA or AEZC abend will occur if you DPL to an AMODE(24) program!
 - Define your own mirror transaction with TASKDATALOC(BELOW)
 - Change the application to be AMODE(31)
- Change to the COMMAREA location on EXEC CICS XCTL PROGRAM()
 - Prior to CICS TS V5.1
 - COMMAREA on XCTL always copied to 24-bit
 - CICS TS V5.1
 - COMMAREA on XCTL remains in 31-bit
 - *Copied to 24-bit or 31-bit storage depending on target program*
 - Same behaviour as EXEC CICS LINK PROGRAM()

Notes...

The CICS-supplied mirror transactions use 31-bit storage (above 16 MB but below 2 GB). If an EXEC CICS LINK command is issued via DPL to an AMODE(24) application, an AEZA or AEZC abend will occur. To avoid this situation, do one of the following:

1. Define your own mirror transaction that uses 24-bit storage. For example, you can copy a CICS-supplied mirror transaction, then specify the TASKDATALOC(BELOW) attribute.
2. Modify the application so that it is AMODE(31) and update the appropriate program definition.

Communication areas (COMMAREAs) used with an XCTL command, when the receiver is AMODE(31) will now reside in 31-bit storage not 24-bit storage.



Further 24-bit VSCR

- 24-bit Virtual Storage Constraint Relief
 - User Exit Global Work Area
 - New GALLOCATION parameter on the ENABLE PROGRAM command
 - *Specifies the location of the storage that CICS provides as a global work area for this exit program. You must also specify the GALENGTH option to create the global work area.*
 - *LOC24 → The global work area is in 24-bit storage.*
 - *This is the default location.*
 - *LOC31 → The global work area is in 31-bit storage.*
- IPCS VERBX DFHPD680 runs RMODE(ANY)



Notes ...

When you enable a global user exit program or task-related user exit program, you can now select the location for the storage that CICS provides as a global work area for the exit. The global work area can be in 24-bit storage or in 31-bit storage.



Remove 32K restriction on MQ DPL Bridge message size

- CICS-WebSphere MQ DPL Bridge
 - Supports a Channel/Container based interface
 - Flexible, not restricted to 32 KB
 - One request container, one response container
 - Transaction CKBC, defined in group DFHMQ
 - Place request WMQ message into DFHREQUEST container
 - Link with channel DFHMQBR_CHANNEL
 - Target can return a response in DFHRESPONSE container



Notes ...

The CKBC category 2 transaction code is used by the CICS-WebSphere MQ DPL bridge so data can now be passed as a container. Containers are not restricted to the maximum 32 KB size of a COMMAREA, so this enhancement gives users greater flexibility.

To use channels and containers, you must specify either the new CKBC transaction code, or your own transaction code modeled on CKBC, to run program DFHMQBP3. This program passes and receives data by using the DFHMQBR_CHANNEL channel and the DFHREQUEST and DFHRESPONSE containers.

You can continue to use existing CICS-WebSphere MQ DPL bridge facilities without changing your programs or configuration. For example, to use COMMAREA, either allow the transaction code to default to CKBP, or specify CKBP (or a transaction code modeled on CKBP), to run program DFHMQBP0 and pass and receive data by using the COMMAREA.



VSCR – Greater Use of 64-bit Storage

- 31-bit Virtual Storage Constraint Relief
 - Reduce pressure on above the line storage
 - Reduce above the line Short-on-storage situations
 - Provide for greater capacity for workload growth
- Greater Use of 64-bit Storage
 - CICS Domain control blocks moved from 31-bit to 64-bit
 - Console Queue Domain – Selected storage subpools
 - Loader Domain – Selected storage subpools
 - *Implications for XPI using PROGRAM_TOKEN, NEW_PROGRAM_TOKEN*
 - Storage Manager Domain – Additional control blocks moved into 64-bit
 - New components exploiting 64-bit storage
 - e.g. Managed Platform (Policies), Application Context data
 - CICS ***non-LE assembler*** AMODE(64) support

Notes ...

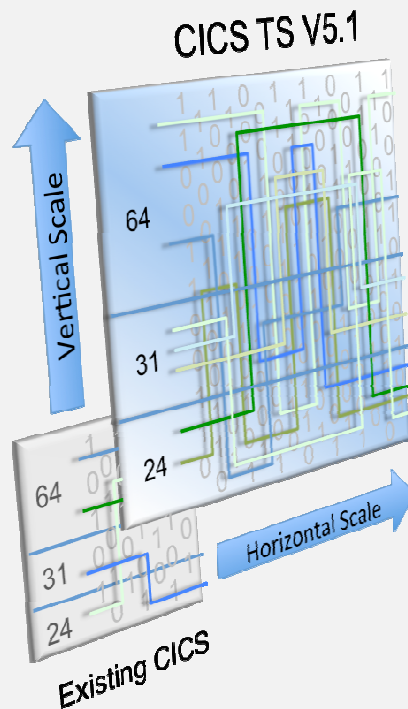
Several CICS® facilities now use 64-bit (above-the-bar) storage in the CICS region, in place of 31-bit (above-the-line) storage. These changes increase the virtual storage available for your existing and new applications to operate in 31-bit storage.

The CICS runtime environment is improved in the following ways:

- The need for additional CICS regions to resolve demands on 31-bit storage is reduced.
- Problems caused by short-on-storage conditions in 31-bit storage are avoided.
- CICS regions can carry out simultaneous processing of a greater number of tasks that use these CICS facilities.
- CICS trace**
 - CICS now uses 64-bit storage for the internal trace table, so some messages and trace points are changed.
 - The default size of the transaction dump trace table, set by using the TRTRANSZ system initialization parameter, is increased from 16 KB to 1024 KB.
- Console queue processing**
 - CICS now uses 64-bit storage for the console queue processing trace table and the console queue transaction entry table. These tables were previously in 31-bit storage taken from the ECDSA.
- Storage allocation control blocks**
 - CICS now uses 64-bit storage for the storage element descriptor (SCE) and free storage descriptor (SCF) control blocks, which control storage allocation. Use of 24-bit and 31-bit storage is reduced, especially in systems with a lot of storage allocation activity, for example, systems with subpools that keep an element chain and that have many small records.
- Loader control blocks**
 - CICS now uses 64-bit storage for the Active Program Element (APE), Current Program Element (CPE), and CSECT descriptor control blocks in the loader domain. These control blocks were previously in 31-bit storage, and could occupy a significant amount of storage.
 - To provide access to the 64-bit storage, the size of the tokens used on the PROGRAM_TOKEN and NEW_PROGRAM_TOKEN options on the XPI calls ACQUIRE_PROGRAM, DEFINE_PROGRAM, and RELEASE_PROGRAM has increased from 4 bytes to 8 bytes. You must change and recompile global user exit programs that use these options. Exit programs that do not use the PROGRAM_TOKEN or NEW_PROGRAM_TOKEN option are not affected.



Greater capacity



- Simplified configuration and increased max. task limit
- Enhanced Instrumentation
- Java 7 support
- OTE and threadsafety
- VSCR and 64-bit exploitation
- Access to 64-bit storage from assembler programs



64-bit CICS Application Support for “big data”

- 64-bit CICS Assembler Application Support – AMODE(64)
 - **AMODE(64) Non-Language Environment Assembler Only!**
 - Provides application support to access large data objects
 - Application can cache large amounts of data above the bar
 - Application must copy data into 31-bit storage if used on CICS API
 - For example as FROM data when writing to a file
 - Application can use containers to pass data
 - CICS keeps the container data in 64-bit storage
 - CICS passes the data to applications in 31-bit/64-bit storage as appropriate



Notes ...

The use case for AMODE(64) assembler applications is to allow caching of user data in 64-bit storage.

Data can be passed to other AMODE(64) assembler applications via containers, or to AMODE(31) applications via containers.

For CICS API requests, the referenced data must be 31bit storage, for example if the data was to be written to a VSAM file



64-bit CICS Application Support – API

- 64-bit CICS Assembler Application Support – AMODE(64)
 - Only the CICS Command Level Programming Interface is supported!
 - No support for CICS Resource Manager APIs
 - *e.g. DB2, WebSphere MQ, IMS DBCTL, etc, ...*
 - 64-bit CICS API
 - CICS Managed 64-bit Storage – CICS, USER, SHARED
 - *EXEC CICS GETMAIN64 and FREEMAIN64 for 64-bit storage*
 - *Task and Shared Storage*
 - Channels and Containers
 - *EXEC CICS GET64 CONTAINER*
 - *retrieves data from a named container into 64-bit storage*
 - *EXEC CICS PUT64 CONTAINER*
 - *places data from 64-bit storage in a named container*

Notes ...

CICS TS supports non-Language Environment (LE) assembler language programs that run in 64-bit addressing mode (AMODE(64)), thus providing 64-bit application support to access large data objects.

New API commands, a new CICS-supplied procedure (DFHEGTAL) and new CICS executable modules are supplied to provide 64-bit application support. CICS storage manager, program manager, loader domain, CICS-supplied macros, CECI, and CEDF are changed to provide 64-bit application support. New dynamic storage areas (DSAs) are available in 64-bit storage.

The following commands can be issued in AMODE(64) and provide the ability to cache data in 64-bit storage:

EXEC CICS GETMAIN64

EXEC CICS FREEMAIN64

EXEC CICS GET64 CONTAINER

EXEC CICS PUT64 CONTAINER

Requests to resource managers are **not** supported in AMODE(64) assembler programs.



64-bit CICS Application Support – considerations

- AMODE(64) CICS Application Program Support
 - Use of existing CICS API switches to AMODE(31)
 - EXEC CICS LINK, LOAD, XCTL, RETURN to/from any AMODE
 - Changes to existing EXEC CICS API commands
 - EXEC CICS LOAD PROGRAM() ENTRY()
 - EXEC CICS INQUIRE PROGRAM() ENTRYPOINT()
 - EXEC CICS ASSIGN ASRAREGS64() and ASRAPSW16()
- *There is no performance advantage to be gained with AMODE(64) unless you update the application to exploit 64-bit virtual storage*

Notes ...

All CICS API commands other than GETMAIN64, FREEMAIN64, GET64 CONTAINER and PUT64 CONTAINER will switch to AMODE(31) to process the request.

Amode(64) applications can interact with other AMODE CICS applications using CICS program control commands

The ENTRYPOINT option of the INQUIRE command is changed to support non-LE AMODE(64) assembler programs, indicating the addressing mode of the load module:

AMODE(24): bit 0 is 0 and bit 31 is 0

AMODE(31): bit 0 is 1 and bit 31 is 0

AMODE(64): bit 0 is 0 and bit 31 is 1

LOAD PROGRAM ENTRY similarly reflects the addressing mode.



64-bit CICS application support Recommendations and Restrictions

- Recommend using Relative Addressing – default for AMODE(64)
 - But this is not mandatory
- EXEC CICS API Commands that are Not Supported
 - EXEC CICS HANDLE/IGNORE CONDITION
 - Use *RESP/RESP2*
 - EXEC CICS HANDLE ABEND LABEL
 - *EXEC CICS HANDLE ABEND PROGRAM()* is supported
- AMODE(64) Assembler Programs are NOT supported as:
 - Global or Task User Exit Programs (GLUEs and TRUEs)
 - User Replaceable Programs (URMs)
- Make sure you use the correct EXEC API Stub – DFHEAG
- FILEA samples illustrate AMODE(64) coding



Notes ...

We recommend AMODE(64) applications use relative addressing.

Handle and ignore condition commands are not supported, so use RESP and RESP2 on the commands instead. Likewise HANDLE ABEND LABEL is not supported, but HANDLE CONDITION PROGRAM is supported.

Support for non-LE assembler programs running AMODE(64) is provided for application programs only. No support is provided for Global or task related user exits or user replaceable modules.

CICS provides a new stub DFHEAG which must be linkedited with AMODE(64) applications.

Most of the FILEA sample programs have been updated to AMODE(64), e.g. DFH\$AALL and DFH\$ABRW, with the exception of DFH\$AREP which demonstrates HANDLE CONDITION LABEL

Scalability – z/OS benefits

- z/OS R13
 - Minimum z/OS release requirement for CICS TS V5.1
- SDUMP Performance
 - Provided in z/OS R12 & above
- z/OS JCL DD statement – SPIN parameter
 - SPIN= on DD card
 - Use the SPIN parameter to specify that the output for the SYSOUT data set is to be made available for processing
 - CICS 24x7
 - Can be used to make the Transient Data Message logs available for processing without the need to shutdown and/or close/deallocate the transient data queue datasets.



Notes ...

z/OS 1.13 is the minimum level required to run CICS TS 5.1. CICS TS will not initialise on anything earlier.

CICS can benefit from enhancements to SDUMP performance that were made in z/OS 1.12

z/OS 1.13 provides enhanced support for use of SPIN on the DD card for spool datasets which allows capture (for example, of MSGUSR data) and reuse without having to bring down CICS.

LE APAR PM57053 reduces the amount of 24bit storage required by LE in a CICS environment.

Summary

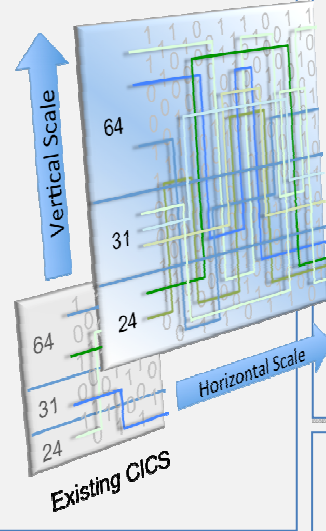


CICS TS V5.1 - Driving Operational Efficiency



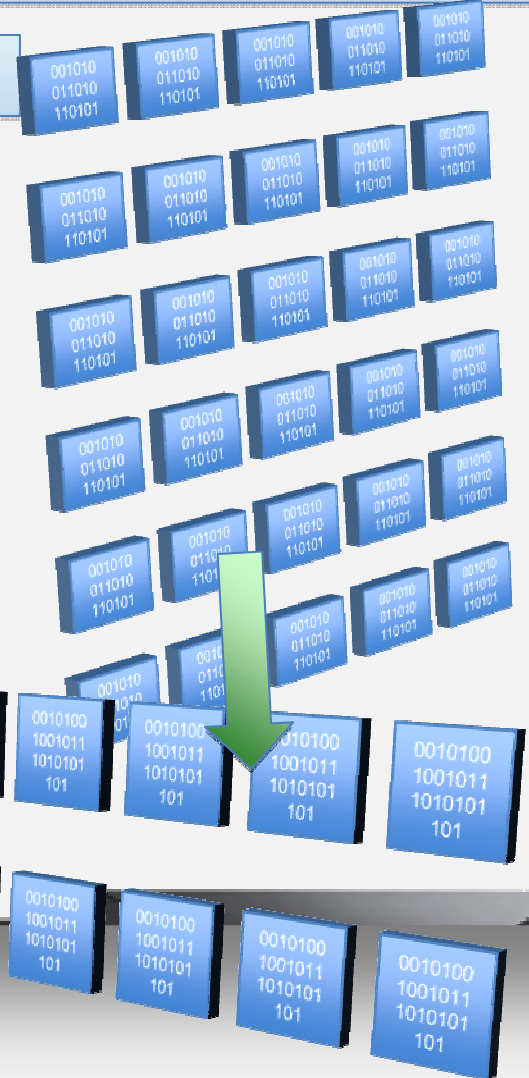
Greater Capacity

CICS TS V5.1



- Doubling of the MAXTASK limit to 2,000
- Increased 64-bit and reduced 24-bit storage usage
- Greater parallelism from threadsafe API and SPI
- Greater system parallelism through optimized TCB usage
- Performance improvements from 64-bit Java 7
- Greater access to 64-bit storage from Assembler programs

- Better reflect current best practices with updated and simplified defaults
- Calculation of the actual and potential use of specialty processors
- CICS-MQ DPL Bridge message size > 32K



Any Further Questions ?



I ♥ CICS

Complete your sessions evaluation online at SHARE.org/SFEval