

2012

# IBM System z Technical University

Enabling the infrastructure for smarter computing

## CICS TS V5.1 Technical Overview

Operational efficiency and service agility...

...with cloud enablement

**Session zAI07**

John Tilling  
CICS Technical Planning & Strategy  
IBM UK Laboratories



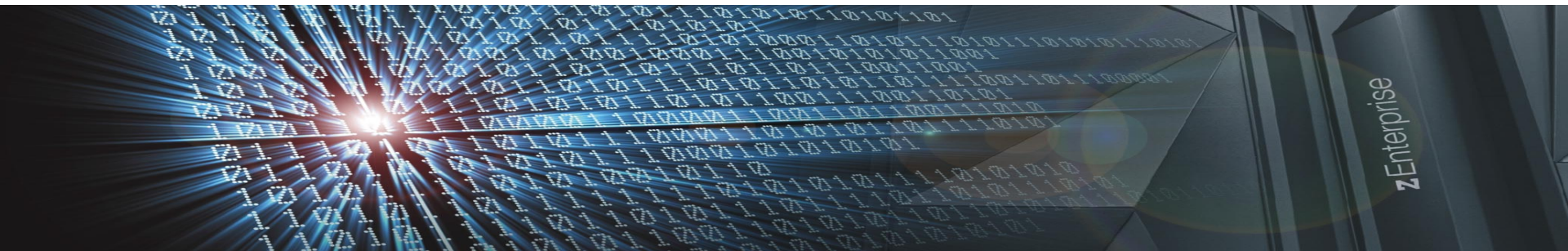
## CICS Transaction Server – Business perspective

- CICS TS is the world's premier enterprise class transaction processor
- CICS TS is designed to enable customers to create and maintain a competitive advantage
- CICS Transaction Server continues to deliver innovative and essential technology to enable customers to deliver on their business imperatives, now and into the future

*Large scale business critical workloads, some > 1bn transactions per day*

*Large installations with mixed workloads continue to support core business*

*CICS WebServices are the most widely adopted CICS feature in the last 10 years*



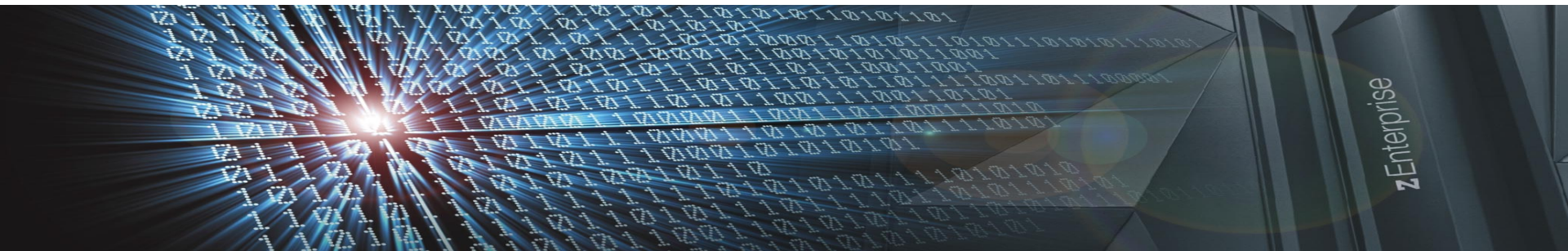
## Current challenges

- Critical importance is that operational costs are kept as low as possible
- Increased pressure to be able to deliver results faster and more efficiently
  - Current operational practices often inhibit the speed of delivery
- How do you keep pace with rapidly changing technology and trends?

*Operation Efficiency:  
Controlling cost*

*Service Agility:  
Deliver results faster*


*Cloud Enablement:  
A long term investment*



## Announcing the new CICS TS V5.1 release

### 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

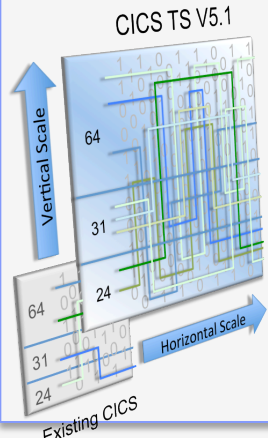
... *with Cloud Enablement*

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 5.1 - Driving Operational Efficiency



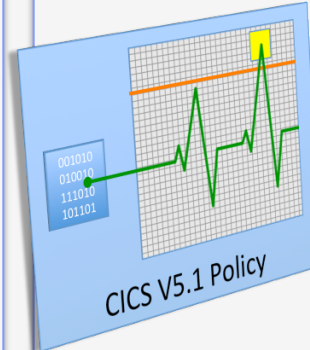
### Greater Capacity



*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*

## Notes ...

The challenge of Driving operational efficiencies is to be addressed in four ways:

- Greater capacity is delivered through significant vertical and horizontal scalability enhancements.
- Managed operations are provided through the introduction of policies, which deliver automated control over critical system resources.
- Increased availability enhancements reduce the need for planned outages.
- Deeper insight into performance and compliance information can help to tune and audit system usage.

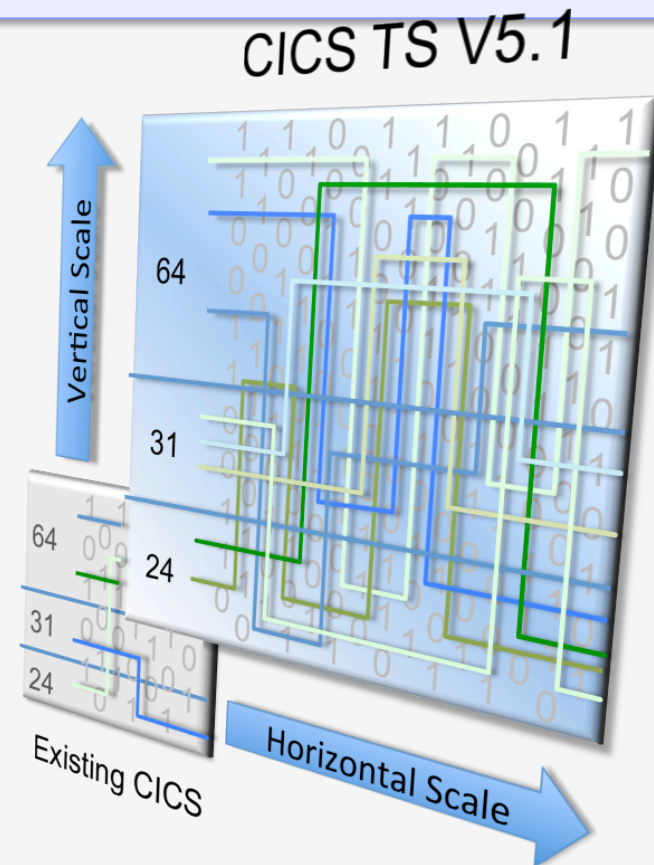
## Driving operational efficiencies - Greater capacity



## Greater horizontal and vertical scaling

*New and improved capabilities provide...*

- Greater single region scalability by doubling the MAXTASK limit to 2,000
- Greater usage of 64-bit storage and reduced usage of 24-bit storage
- Greater application parallelism through threadsafe API and SPI extensions
- Greater system parallelism through optimized Task Control Block (TCB) usage
- Greater Java performance improvements from 64-bit Java 7 support
- Greater access to 64-bit application storage when using Assembler programs



## Notes ...

Greater capacity is delivered through significant vertical and horizontal scalability enhancements. The slide summarizes the enhancements which will be described in more detail over the next few slides.





## Greater capacity

- Max number of concurrent user tasks per region is now 2000
  - MXT minimum increased from 1 to 10
  - Default changed from 5 to 500
  - 24 bit VSCR in particular gives more capacity to a single region
  - Allows consolidation of regions – after appropriate tuning of storage limits
  - New instrumentation for current and max tasks values
  
- Additional CICS domains exploiting 64-bit storage
  - Console Queue Domain
  - Storage Manager Domain
  - Loader Domain
  - New function exploiting 64-bit storage ...
    - e.g. Managed Platform, Application Context
  - CICS requires a MEMLIMIT setting of 6GB or higher

## 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.

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.

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.

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.



## Greater capacity

- 24-bit Virtual Storage Constraint Relief ...
  - Control blocks, Modules, and stack storage moved above the line
    - Syncpoint, Transient Data, Journal Control, ...
  - Transient Data access method buffers – Extrapartition transient data
    - Moved from 24-bit to 31-bit
  - Removed 24 bit storage consumption by CICS supplied transactions
    - Redefined with TASKDATALOC(ANY)
      - CEMT, CEOT, CESN, CESF, CETR, CMSG, CRTE, ...
      - CWTO, CIEP, CSNC, and the Mirror transactions ...
      - CEDF and CECL processing

## Notes ...

The following CICS infrastructure items now use 31-bit storage (above 16 MB but below 2 GB) in place of all, or some, of the 24-bit storage (below 16 MB) that was used in previous releases:

- Communication areas (COMMAREAs) used with an XCTL command, when the receiver is AMODE(31)
- Processing for transient data EXEC CICS application programming commands, wherever possible
- Processing for journaling EXEC CICS application programming commands
- Processing for transaction dumps and for the EXEC CICS DUMP TRANSACTION command
- Access to QSAM to write data to extrapartition transient data destinations (as a result of this change, subpool DFHTDG24 has been removed)
- Exit program link (EPL) control blocks
- Console write-to-operator transaction (CWTO)
- Error message writer
- Sync point and backout processing
- Shutdown processing
- CICS execution diagnostic facility (CEDF)
- CICS command-language tables for command interpreter (CECI) and other functions
- Processing for function-shipped DL/I calls
- Enable and disable functions for work queues for interregion communication
- Opening of the z/OS® Communications Server ACB
- ECI over TCP/IP listener task (CIEP)
- Interregion communication control task (CSNC)
- Transaction environment transaction (CSXM)
- Global work area for the CICS DB2® task-related user exit
- Global work area for the CICS IMS™ DBCTL task-related user exit
- Global work area for the CICS WebSphere® MQ task-related user exit
- Enable function for global user exits for emergency restart processing
- Mirror transactions (CEHP, CEHS, CPMI, CSHR, CSMI, CSM1, CSM2, CSM3, CSM5, and CVMI)



## Greater capacity – OTE enhancements

- Commands that access Transient Data (TD) are now threadsafe
  - EXEC CICS READQ TD, WRITEQ TD, and DELETEQ TD
  - CICS system parameter TDSUBTASK obsolete
  - TD Global User Exits must be threadsafe
  - TD function shipping over IPIC will use an Open TCB
  
- More SPI commands are now threadsafe
  - EXEC CICS SET TASK
  - INQUIRE and SET TRACEDEST / TRACEFLAG / TRACETYPE
  
- Removed TCB switch for Java applications accessing DB2
  - Required PTFs for DB2 V9 (UK78500) and V10 (UK78499)
  
- Increase the potential CICS program LOAD capacity
  - Reduced contention for the single CICS RO TCB
  - New Statistics on RO TCB program load requests and load time

## Notes ...

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 IPIC 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 XTDREQ, XTDEREQC, XTDREQ, XTDIN, and XTDOU must be coded to threadsafe standards and defined to CICS as threadsafe.

Existing SPI commands made threadsafe in this release:

INQUIRE TRACEDEST, INQUIRE TRACEFLAG, INQUIRE TRACETYPE

SET TASK

SET TRACEDEST, SET TRACEFLAG, SET TRACETYPE

New SPI commands that are threadsafe:

INQUIRE EPADAPTERSET, INQUIRE EPADAPTINSET, SET EPADAPTERSET

New API commands that are threadsafe:

FREEMAIN64, GETMAIN64, GET64 CONTAINER, PUT64 CONTAINER

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, applications can now access JDBC and SQLJ from a T8 TCB instead of forcing a move to an L8 TCB.

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 loads 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 that contains programs needs to be opened or closed.



## Greater capacity - 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 FROM data must be 31bit storage, for example if the data was to be written to a vsam file





## Greater capacity - 64-bit CICS Application Support for big data

- 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
  - 31 bit CICS API and SPI
    - CICS switches to amode(31) for existing CICS API & SPI

## Notes ...

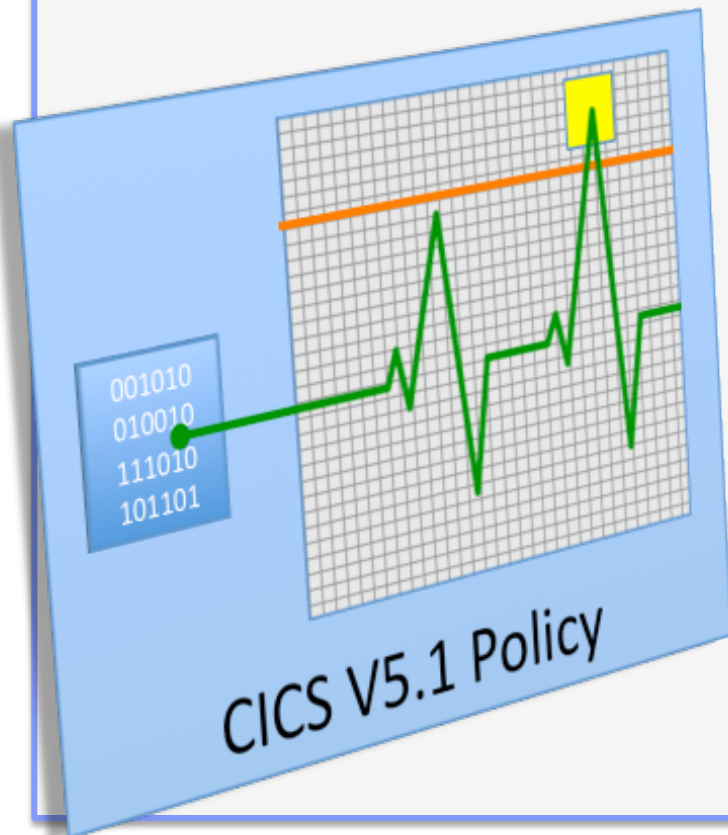
CICS Transaction Server supports non-Language Environment (LE) assembler language programs that run in 64-bit addressing mode (AMODE(64)), thus providing 64-bit application support to access large data objects.

New API commands, a new CICS-supplied procedure 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.

Existing CICS API will execute in amode 31 if called from a 64 bit assembler program.

There is no support to call resource managers (DB2, IMS, MQ) from amode 64 assembler programs.

## Driving operational efficiencies - Managed operations

*Control critical resource thresholds with policies**Now you can...*

- Create policies to deliver automated control over critical system resources
- Set data access thresholds on SQL or file access requests by a user task
- Set program loop thresholds on EXEC LINK requests by a user task
- Set storage request thresholds on storage used or GETMAIN requests
- Set CPU time thresholds on the amount of processor time used by a user task
- Manage policy breaches by issuing messages, abending tasks, or creating events

## Notes ...

Managed operations are provided through the introduction of policies, which deliver automated control over critical system resources

A new, dynamic policy-based management capability is introduced in support of both applications and platforms. Policies enable the behavior of applications and platforms to be managed by determining whether tasks running as part of a platform, as an application, or as types of operation within an application, exceed certain predefined thresholds.

Task thresholds can be set based on data access requests, storage usage, program loops and processor time used. For example, a threshold could be defined based on the amount of above-the-line storage used by a task, the number of times a task accesses IBM DB2 or a file, or the number of EXEC LINK requests issued by a user task.

After a threshold is exceeded, CICS can issue a message, or abend the task with a specific abend. Additionally, policies can be defined to trigger one or more CICS events, which can in turn initiate other actions.

CICS policies are a declarative way of ensuring that applications and platforms continue to run effectively. A policy can be applied to any combination of applications and platforms. Additionally, policies can also be deployed into a single region, independently of defining a platform. Policies are applied dynamically during production operations.

In summary you can define threshold conditions to limit the following resources:

- The amount of task or user storage below the line (24-bit), above the line (31-bit), or above the bar (64-bit).
- The number of requests for task or user storage below the line (24-bit), above the line (31-bit), or above the bar (64-bit).
- The number of program link requests.
- The number of database requests (SQL commands).
- The number of file access requests, that is: Read, Read update, Write, Rewrite, Delete, Start browse, Read next, or Read previous requests.
- The amount of CPU time consumed by a task.



## Managed operations - policies

Control resource consumption

Consists of one or more rules in an XML document created using CICS Explorer and deployed via a CICS bundle

**Threshold:** CPU, storage, database access, ...

**Action:** message, event, abend

**Create Policy Definition**

**Add a rule**  
Add a policy rule to define what action is taken when a task exceeds a specified condition.

**Policy Information**  
Name:  (is also the bundle part name)  
Description:

**Rule Information**  
Name:   
Description:

What is the condition that triggers the rule?  
Type:   
Item:

Operator:  Value:  Unit:

What action should be taken when the rule's condition is exceeded?  
 Issue message DFHMP3001  
 Emit event to  
 EP Adapter   
 EP Adapter Set   
 Abend task with abend code AMPB

Open Editor

## Notes ...

Policies contain one or more rule types with associated thresholds and actions. You can deploy policies to monitor the resource utilization of a user task, and to automatically respond when resource usage exceeds the thresholds you define. When a threshold is exceeded, you can specify that CICS issues a message, emits an event, or abends the task. In this way, excessive resource usage and looping and runaway transactions can be detected and dealt with appropriately.

The following actions can be performed when a threshold is exceeded:

- Issue message DFHMP3001, which is the default policy action.

- Abend the task. Either the default abend code AMPB or a user-specified abend code is issued (by editing the xml).

- Emit an event.

A condition and action pair make up a policy rule, and one or more policy rules can be defined within a policy. A policy is defined in a CICS bundle and a CICS bundle can consist of one or more policies.



## Managed operations - policies

- CICS bundles that contain policies can be packaged into application bundles,
  - Deployed to a platform when you install the application
  - Alternatively, CICS bundles that contain policies can be defined in the CICS CSD or CPSM data repository, and installed into any CICS region
  
- Policies are deployed to a specific scope
  - an application scope, or
  - an operation scope within the application, or
  - a platform scope, or
  - no scope - meaning it is scoped to the CICS region
  
- Policy rules for an application viewable via CICS Explorer

## Notes ...

You define a policy in a CICS bundle using the CICS Explorer®. CICS bundles that contain policies can be packaged into application bundles, so that they are deployed to a platform when you install the application. Alternatively, CICS bundles that contain policies can be defined in the CICS CSD or CICSplex® SM data repository, and installed into any CICS region.

Policies are deployed to a specific scope. The scope can be an application scope, an operation scope within the application, a platform scope, or no scope. If the policy is not deployed to a scope it is scoped to the CICS region. When a policy is deployed with an application scope it applies to all user tasks within the platform that have the matching platform, application, and application version information in their application context. When a policy is deployed with a platform scope it applies to all user tasks within the platform that have the matching platform in their application context. When a policy is deployed with an operation scope, it applies only to user tasks that also match the operation.



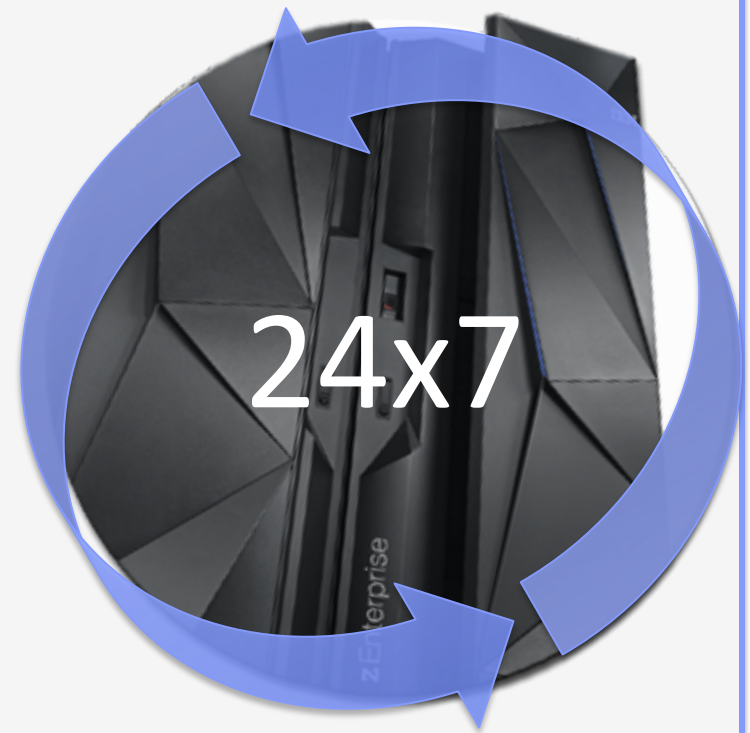
## Driving operational efficiencies - Increased availability



## *Reduce the need for planned downtime*

*New and improved capabilities enable you to...*

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



## Notes ...

Increased availability enhancements reduce the need for planned outages. The slide summaries the enhancements which will be described in more detail over the next few slides.



## Increased availability enhancements

- Dynamically install the CICS TYPE 3 supervisor call (SVC) using new utility DFHCSVCU
  - No need to IPL z/OS therefore more flexible and faster to install CICS
- SSL certificates and environment can be refreshed online for improve 24x7 operation
  - New **PERFORM SSL REBUILD** command
    - After updating key ring with new certificate
    - After LDAP server is restarted
- IPIC heartbeat messages to systems that can respond to them
  - This ability to respond allows the integrity of the connection to be assessed and maintained by CICS regions through periods of inactivity.

## Notes ...

A new utility, DFHCSVCU, is provided to dynamically install the CICS Type 3 supervisor call (SVC) without the requirement to re-IPL the z/OS system. This program simplifies the CICS installation process so that less time is required when you next upgrade your systems. Note: You can use the DFHCSVCU utility program only if you have the required authorization.

System administrators can use the new PERFORM SSL REBUILD command to refresh the certificates used by a CICS region for SSL handshakes. When a CICS region that uses SSL starts, an SSL environment is established for the CICS region. z/OS System SSL, the Cryptographic Services component of z/OS, manages the SSL environment. The SSL environment includes a cache containing copies of the certificates in the RACF key ring for the CICS region.

Before CICS® TS for z/OS®, Version 5.1, when you added a new certificate to the key ring or renewed a certificate that was about to expire, you had to restart the CICS region in order to rebuild the cache of certificates and make the updated information accessible to the CICS region. Now, when you carry out these certificate maintenance tasks, you can issue the PERFORM SSL REBUILD command to rebuild the SSL environment and make z/OS System SSL repopulate the cache of certificates. The CICS region continues to run, and new SSL handshakes or sessions that begin in the CICS region after the command completes use the refreshed certificate information.

The PERFORM SSL REBUILD command starts a new local cache of SSL sessions for the CICS region. A cache of SSL sessions held at sysplex level for multiple CICS regions is not affected. The command also refreshes the bind information that is held for the LDAP server that stores certificate revocation lists (CRLs), but it does not refresh the certificate revocation lists on the LDAP server.

You can now automatically send IPIC heartbeat messages to systems that can respond to them. This ability to respond allows the integrity of the connection to be assessed and maintained by CICS regions through periods of inactivity.

IPIC connections use TCP/IP sockets which can be routed across wide area networks that route requests through firewalls which permit or deny network transmissions. These firewalls are typically configured with policies that block TCP/IP packets if no messages are transmitted over a particular socket in a specific period. If this happens, CICS transactions can hang if requests are sent across a dormant connection. CICS can now maintain IPIC connections through periods of inactivity by maintaining a heartbeat that uses internal IPIC messages. You cannot directly influence the frequency of this heartbeat operation.



## Increased availability enhancements

- CICS TS 5.1 & CICS VR 5.1 support replication logging
  - New VSAM cluster attribute – results produce consolidated log records for software replication
  - IBM Statement of Direction:

“IBM intends in the future to enhance the IBM Geographically Dispersed Parallel Sysplex (GDPS)/Active-Active continuous availability solution by providing support for software replication of Virtual Storage Access Method (VSAM) data for active-standby and active-query configurations. IBM plans to provide such support for data replication for VSAM data updated by applications running in CICS and offline in batch mode, using log data provided by IBM CICS Transaction Server for z/OS V5.1, or later, and IBM CICS VSAM Recovery for z/OS V5.1, or later.”
- CICS TS V5.1 introduces a number of enhancements to make operations easier and more transparent
  - the default values for several CICS SIT parameters are changed to adopt best practice
  - or are removed altogether where CICS is now able to automatically make adjustments at run time. eg MAXOPENTCBS, MAXXPTCBS, MAXJVMTCBS

## Notes ...

CICS® TS 5.1 provides replication logging capability in support of the IBM® GDPS/Active-Active (GDPS/AA) availability solution which IBM intends, in the future, to enhance to support replication of VSAM data for active-standby and active-query configurations. Currently GDPS/AA provides software replication of DB2® and IMS™ data between geographically dispersed sysplexes. The replication logging in CICS TS is intended to be used by GDPS/AA to provide replication of VSAM data updated under the control of CICS TS and compliments replication logging in CICS VR 5.1 which provides the same support for batch jobs.

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.

## Driving operational efficiencies - Deeper insight



### *Extend performance and compliance information*



#### *New and improved capabilities enable...*

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

## Notes ...

Deeper insight into performance and compliance information can help to tune and audit system usage. The slide summaries the enhancements which will be described in more detail over the next few slides.





## Deeper insight

- SPI commands (SET, PERFORM, ENABLE, DISABLE, and RESYNC) now audited to the CICS CADS TD destination
  - DFHAP1900 *date time applid netname userid transid command* RESP(response) RESP2(n)
    - Issued after command completes, for example...
      - DFHAP1900 09/17/2012 09:47:53 IYK2Z2G1 IYCWT195 CICSUSER CEMT SET FILE(FILEA) NOWAIT OPENSTATUS(CLOSED) RESP(NORMAL) RESP2(0).
      - DFHAP1900 09/17/2012 09:48:03 IYK2Z2G1 NONE CICSUSER CONL SET FILE(FILEA) BUSY(NOWAIT) ENABLESTATUS(ENABLED) OPENSTATUS(OPEN) RESP(NORMAL) RESP2(0).
- Logging on with VERIFY command can now use full RACF verification at least once a day
  - Ensures user IDs are always ready for audit, recorded as being used, and retained in your system
  - Applies to HTTP basic authentication for web support, Webservices & IPIC authentication & use of VERIFY API commands
- Distributed identities are now propagated for START command

Auditors can use the distributed identity associated with started tasks to find the identity of the user

## Notes ...

Some system programming interface commands are now audited. This feature provides useful information about configuration changes for problem diagnosis. The system programming interface commands SET, PERFORM, ENABLE, DISABLE, and RESYNC are audited. When these commands are issued, message DFHAP1900 is written to the CICS® job log. The message contains information about the parameters used in the command.

You can now specify that a full verification request takes place at least once a day when users log in to the CICS region. This process ensures that user IDs record their most recent usage date for requests made over IP.

When a user logs in to CICS by a method that uses password verification, like the EXEC CICS VERIFY PASSWORD or EXEC CICS VERIFY PHRASE command, instead of a full verification request like the EXEC CICS SIGNON command, the login process is faster. However, with the faster login methods, RACF does not record the login as the last access for the user ID, and does not write audit information for the user ID. User IDs that are always used with the faster login methods can therefore appear to be unused, and could be revoked.

CICS uses password verification for improved performance in the following login processes:

- HTTP basic authentication with CICS web support
- Web services authentication
- IP interconnectivity (IPIC) authentication

You might also have your own login processes that use the EXEC CICS VERIFY PASSWORD or VERIFY PHRASE command.

If you have users who generally use login processes with password verification, you can now specify the system initialization parameter SECIFYFREQ=USRDELAY in the CICS region, to require that CICS makes a full verification request at least once a day for each user ID that is used to log in to the CICS region. The full verification request uses the RACROUTE REQUEST=VERIFYX macro, instead of the RACROUTE REQUEST=EXTRACT macro that is used for password verification. In RACF, the full verification request updates the user statistics for audit purposes, and updates the date and time of last access for the user ID so that it is recorded as being in use.

Operators and auditors can see the distributed identity of the user for started tasks.

Identity propagation was introduced in CICS® TS 4.1. With this feature, the client identity contained a distributed identity (usually an X.500 ID) and a realm to be propagated to CICS in an Extended Context Identity Reference (ICRX). The ICRX was contained in IPIC DPL requests from CICS Transaction Gateway and web services requests. CICS associated the distributed identity with a RACF® user ID using RACMAPs. The ICRX was propagated in CICS in DPL requests using MRO or IPIC. However if the transaction issued a START request, only the RACF user ID was propagated.

In CICS TS 5.1 the ICRX is also propagated for transactions issuing START commands which do not specify a USERID, TERMID or RTERMID. If any of these are specified, CICS does not propagate the ICRX.



## Deeper insight

- **SSL CIPHER code**
  - SSL ciphers used are now recorded in SMF 110 CMF performance class records for better performance analysis
- **zAAP/zIIP Specialty Processor Transaction CPU time**
  - New CMF Performance Class Fields ...
  - Requires z/OS R13 APAR OA38409 and IBM System z9 or later
- **New regular status messages are issued during startup if VSAM RLS data sets require lost locks recovery processing**

## Notes ...

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 DFHSOCK 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.

You can use the new CPUTONCP and OFFLCPUT fields in the DFHTASK performance class group for a transaction to calculate the processor time that a task spends on a zIIP or zAAP specialty processor. These fields can 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 available only 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.

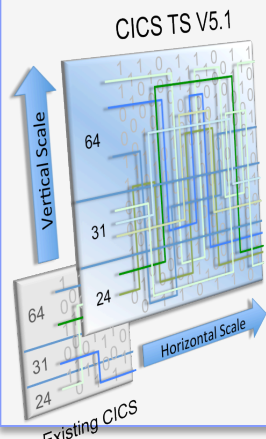
CICS® now provides regular status updates while lost locks recovery is taking place. A new message displays the number of data sets that have completed lost locks recovery, and the total number of data sets that require lost locks recovery. These regular updates ensure that users are aware of the progress that is being made during the time of the recovery.

The lost locks condition is created when a coupling facility lock structure fails and cannot be rebuilt by VSAM. The lost locks condition can occur only for data sets opened in RLS mode.

## CICS TS 5.1 - Driving Operational Efficiency



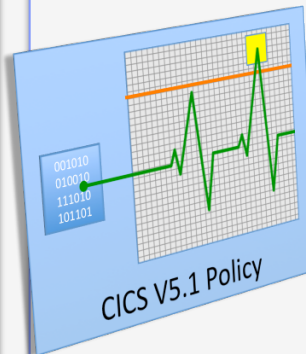
### Greater Capacity



- Doubling 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



### 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 to be stored in the performance records
- Calculate the actual and potential use of specialty processors
- Regular status updates provided while lost locks recovery is taking place

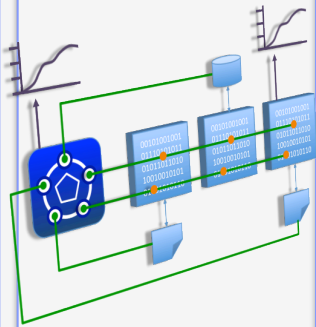
## Notes ...

This slide lists all the enhancements made in CICS TS 5.1 to address the challenge of Driving operational efficiency, grouped under the four key areas.

## CICS TS 5.1 - Increasing Service Agility



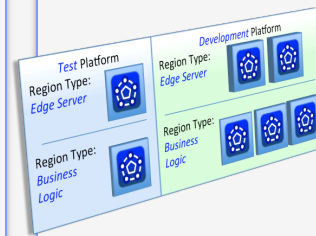
### First-class Applications



*Create agile services from existing assets*



### First-class Platforms



*Create agile service delivery platforms*



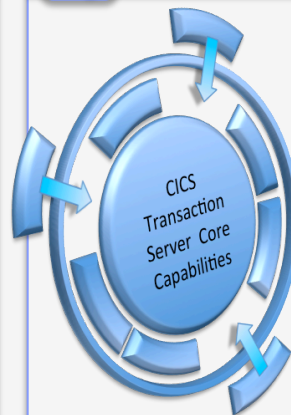
### Modern Interfaces



*Build rich web experiences for critical applications*



### Foundational Enhancements



*Extend core capabilities*

## Notes ...

The challenge of Increasing Service agility is to be addressed in four ways:

- First-class applications enable the creation of agile services from existing assets.
- First-class platforms enable the creation of agile service delivery runtimes.
- Modern interfaces enable customers to build rich web experiences for critical applications.
- Foundational enhancements extend the core CICS capabilities in a number of areas.



## Increasing service agility - First-class applications



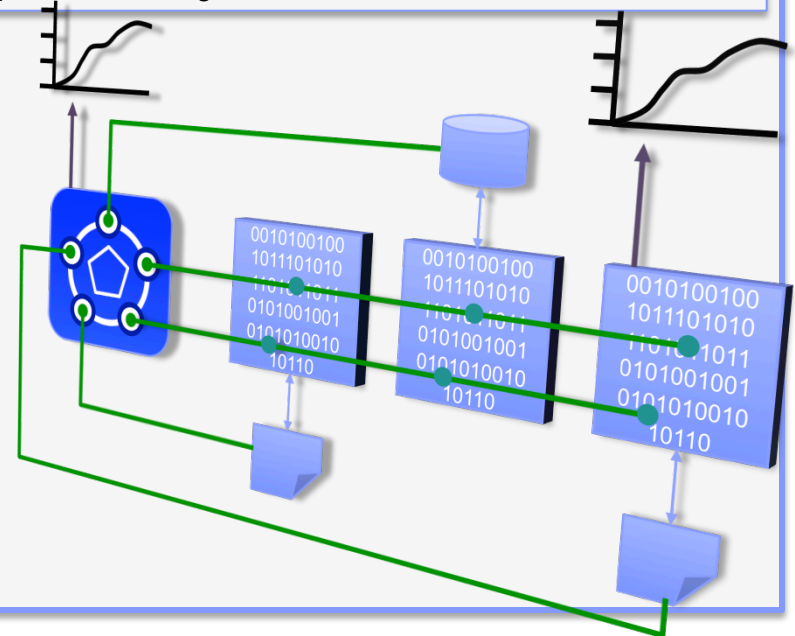
### Create agile services from existing assets

#### By defining an Application you can...

- Combine and manage disparate application resources as a single entity
- Rapidly move versioned applications through development, test and production
- Automate dependency management throughout the application lifecycle
- Ensure rigorous yet flexible provisioning with application bindings
- Measure entire application resource usage for tracking and internal billings
- Dynamically manage applications by applying policies during runtime

#### Measured Service

By defining Application Entry Points, Application context data becomes available in monitoring records to provide a way of measuring how much system resource an application is using



## Notes ...

First-class applications enable the creation of agile services from existing assets. CICS TS V5.1 delivers these capabilities in a way that positions users of CICS for the next transformational era in technology. This moves them towards a service delivery platform for cloud computing.

The large set of disparate but interrelated resources that make up a business application within CICS can be packaged into an application bundle to deploy, manage, and monitor it as a single entity. An application that is defined in this way can be versioned and rapidly moved through development, test and production environments. It provides a single point of control for application lifecycle management.

Dependency management is handled by CICS, and multiple applications can share individual resources. An application describes all its required resources. With one installation action, these resources are created and deployed in the CICS regions where the application requires them. The lifecycle of those resources can be managed together with the application. After an application is no longer required, it can be de-provisioned in its entirety by uninstalling a single application resource. Shared resources that are required by other applications are not affected.

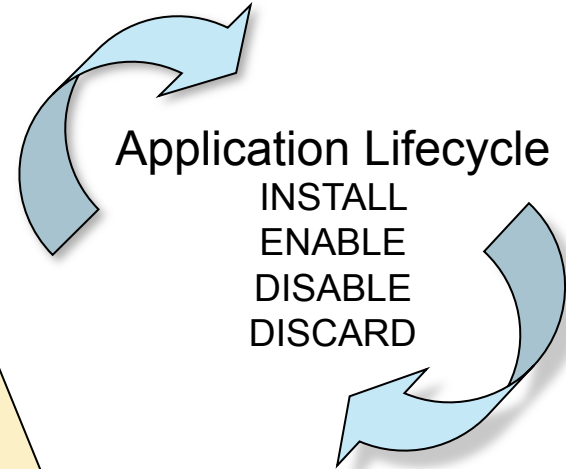
The application binding is used to control this lifecycle, allowing an application to be configured for deployment to a specific platform. This can be through adding additional bundles to the application, which can include policy. Individual parts of the application can be deployed to a subset of the platform without having to change either the application or platform definitions. This enables the same application to be deployed to different platforms, particularly useful in multi-national implementations.

Support for measuring the resource usage of applications is provided. Application entry points can be defined that add application context data to tasks that run in CICS. Application context data is available in monitoring records to provide a way of measuring how much system resource an application is using. This helps users to manage resource usage and process internal charge-back billings. Problem determination can be undertaken in the same manner, starting with the overall application and drilling down to individual tasks and resources.

Applications can be managed dynamically by applying policies during runtime. Policies enable the behavior of applications to be managed by determining whether tasks running as part of an application, or as types of operation within an application, exceed certain predefined thresholds. CICS can take actions when thresholds are breached.



## First-class applications



- A collection of one or more CICS bundles
- Life-cycle as a single entity
- Measure and control resource usage
- Develop in Eclipse/Rational
- Share and promote through SCM

## Application

**Name**  
- org.maw.banking.Loans

**Version**  
- 1.2.1

**Resources**  
- LIBRARY, PROGRAM, TRANSACTION, URIMAP  
- (EVENTBINDING, OSGIBUNDLE, ...)

**Dependencies**  
- DB2CONNECTION, JVMSERVER,  
TCPIPSERVICE, ...

**Entry points**  
- operation: browse, update, ...  
- resource: PROGRAM

## Notes ...

You can package a CICS® application into an application bundle to deploy, manage, and monitor it as a single entity. With an application bundle, you can install and manage the resources for the application in all of the CICS regions in a platform.

- Dependency management is handled by CICS, and multiple applications can share individual resources.
- Applications can be monitored as a logical whole across multiple CICS regions within a platform, helping users to manage resource usage and process internal charge-back billings.
- The ability to drill down to the resource consumption of individual application operations is simplified.
- Problem determination can start with the overall application and drill down to individual tasks and resources.

When you have developed a CICS application, you can use the CICS Explorer® to package it for deployment into a Platform. If your application includes Java™, use the CICS Explorer SDK.

To package an application, you first create a CICS bundle for each application component. In the bundle manifest, the CICS bundle declares all the resources that are needed for the application component, grouping them together and enabling you and CICS to manage them as a single entity. The bundle also declares any dependencies on other resources outside the bundle that must be available in the CICS region where the application component runs.

To group together all the resources for all the application components, you then create an application bundle. The application bundle is a type of management bundle that describes the whole application. The application bundle references the CICS bundles that contain the dependencies and resources for each of the application components. For information about the types of dependencies and resources that you can define in a CICS bundle, see [Defining CICS bundles](#). When you have created the application bundle, you can export it to the platform where you want to run the application. Each platform has a home directory in zFS where applications and policies are stored.

You also create an application binding to define how the application is deployed to a platform. The application binding is an association that maps the bundles in an application to the region types in the target platform. For example, three bundles might be installed into a single CICS region, or one bundle might be deployed to a region type within the platform, and the other two bundles deployed to another region type within the platform. You may also create more complex application bindings to deploy different bundles to different platforms and region types. The application binding is also stored in the home directory for the platform in zFS.



## Application context

- Allows measurement and resource usage control at the application level
- Define the application entry point
  - PROGRAM...
- Associate Task with Application operation
  - PROGRAM LINK...
- The application context follows the application flow from Task to Task & Region to Region across MRO and IPIC connections
- Recorded in monitoring data
  - Includes Platform, Application, Version (major.minor.micro), Operation

## Notes ...

In an application bundle, you can define an application entry point for each of the ways in which your application can be started. You can associate an application entry point with a CICS program that is part of the application. The application entry point specifies the operation that is taking place in the application, such as a read, write, or update operation. When a task that does not have an application context calls a program that has an application entry point, CICS creates an application context that becomes associated with the task, and with any subsequent programs that it calls and tasks that it starts. The application context identifies the platform, the application, the application version, and the operation.

When you view monitoring information about the task in the CICS Explorer or in the monitoring data produced by CICS regions, you can identify the application under which the task is running and see the application context. You can use the information to measure resource consumption by applications (or by particular routes into applications), to use policy-based management for applications, and to relate tasks to specific applications to help with problem diagnosis.

## Increasing service agility - First-class platforms



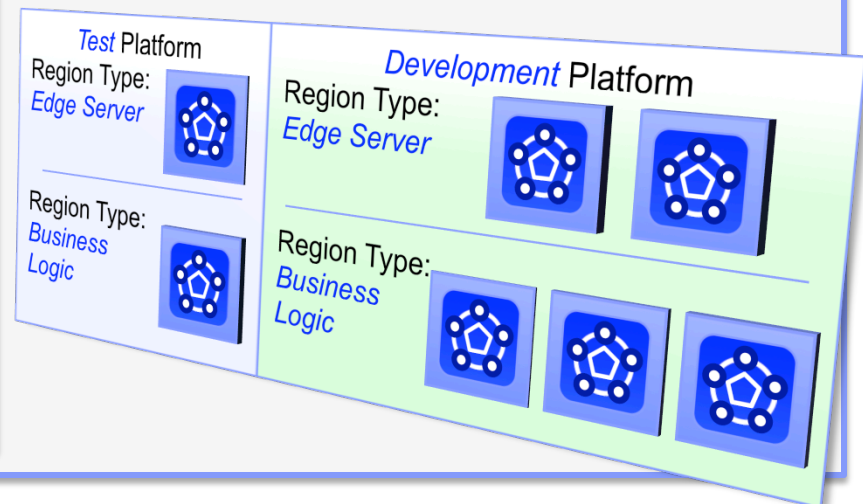
### Create agile service delivery platforms

#### By defining a platform you can...

- Group new and existing regions as platforms for rapid application deployments
- Increase flexibility, by decoupling applications from the underlying CICS region topology
- Increase reliability through automatic resource deployment and validation
- De-provision resources when requested, without requiring any interaction from a system administrator
- Deploy applications to started regions within a platform, without requiring any interaction from a system administrator
- Dynamically manage platforms by applying policies during runtime

#### Greater control through platforms...

- Easily deploy applications onto platforms
- Different platforms can have different policies for the same application
- “How much CPU has this platform consumed?”



## Notes ...

Platforms are new CICS resources that deliver a simple layer of abstraction to decouple applications from the underlying region topology. This helps free the system programming and operations staff from managing at the detailed level of individual regions. Platforms are also a useful construct for managing application separation.

The new CICS platform resource provides a validated environment in which to deploy CICS applications. Newly created and existing regions with similar or identical properties can be grouped together as region types within a platform. Artifacts and resources within an application can be both dynamically deployed to the relevant region types within a platform and removed again when no longer needed.

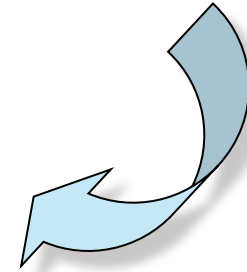
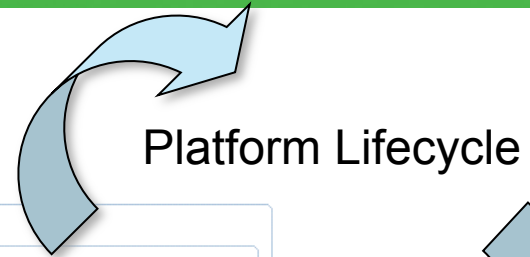
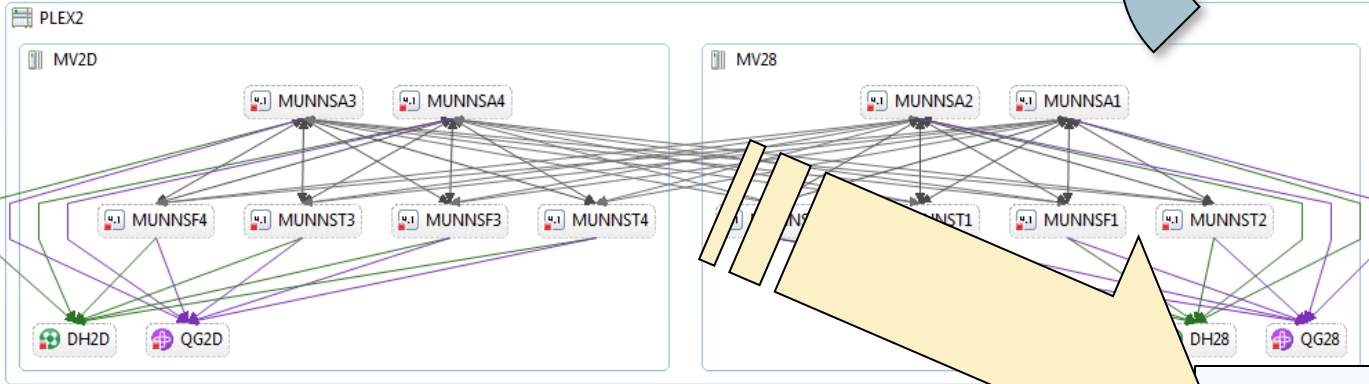
Regions within a platform can be initialized or shut down as necessary to meet fluctuations in capacity requirements. An application can take advantage of these changes without requiring further configuration, and without interruption to the application.

Platforms can be managed dynamically by applying policies during runtime. Policies enable the behavior of platforms to be managed by determining whether tasks running as part of a platform exceed certain predefined thresholds. CICS can take actions when thresholds are breached.





## First-class platforms



### Platform

Region Type

Region Type

Region Type

- Set of one or more region types
- Life-cycle as a single entity
- Hides complexity of underlying topology
- Provides services for Applications
- Control Applications through Policy*

## Notes ...

You can package a CICS® platform into a platform bundle to deploy, manage, and monitor it as a single entity. With a platform bundle, you can install and manage the resources for the platform in all of the CICS regions in a platform.

You can use platforms to help you manage your requirements as a single entity. This has a number of benefits; for example:

- You can manage legal requirements for specific geographies.
- You can manage applications, resources, and policies as a group.
- You can manage system characteristics in one place.
- You can monitor applications or system resources as a group.
- You can monitor platforms as a logical whole, helping users to manage resource usage and process internal charge-back billings.
- You can easily drill down to the resource consumption of individual resources.
- You can undertake problem determination starting with the overall platform and drilling down to individual applications, resources, and policies.

Use a platform instead of a CSYSGRP when you want to manage several groups of systems as a single entity. A CSYSGRP is a method of grouping CICS regions together. A platform provides a point of control for deploying, managing, and monitoring the lifecycle of several groups of systems as a single entity.

## Increasing service agility - Modern interfaces

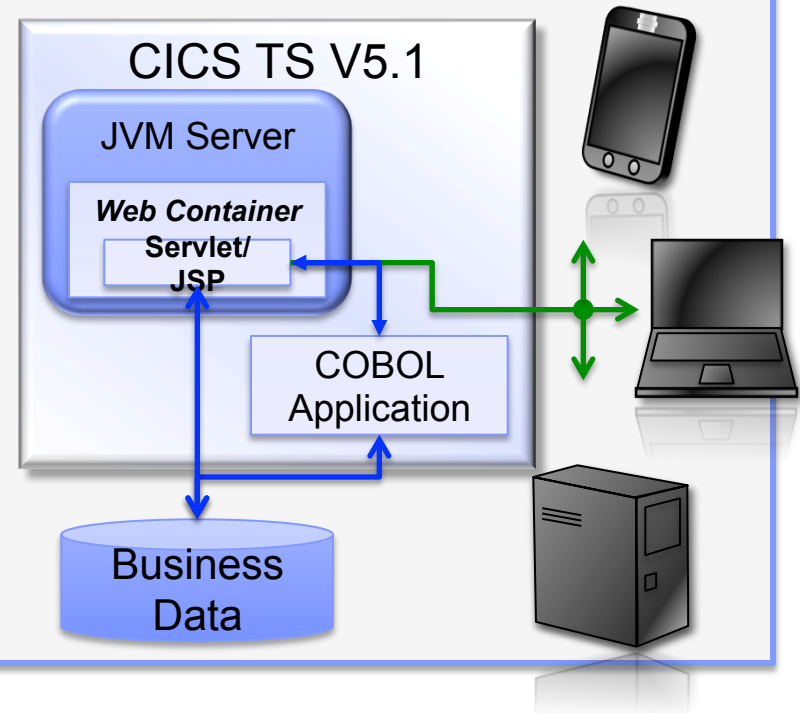


## Build rich web experiences for critical applications

### The web container enables...

- A production-ready web container with CICS qualities of service
- Deployment of lightweight Java servlets and Java Server Pages (JSP)
- Improved performance through local access to CICS applications and data
- Rapid roll-out of interface updates through OSGi-packaged deployments
- Full integration with first-class applications and platforms
- Technology built on the WebSphere Application Server Liberty profile for compatibility

The best connector is no connector!



## Notes ...

New in CICS TS V5.1 is a fast and lightweight production-ready Java web container that provides developers with the rich features of Java servlet and JavaServer Pages (JSP) specifications, combined with CICS qualities of service.

In CICS TS V5.1, significant performance improvements can be delivered, when compared with network-connected web interfaces, by co-locating rich web interfaces with fast local access to your existing CICS applications and data.

The Java web container builds on and extends the OSGi support of the underlying JVM server. This allows for rapid roll-out of interface updates. The web container also supports the deployment of web applications as part of a composite CICS application or platform. This includes full lifecycle management and support for policy thresholds.

This new Java web container is built on WebSphere Application Server Liberty profile technology, which enhances compatibility and portability of web applications and tools in either environment. A wide range of Java development tools can be used to build these web applications, including WebSphere Application Server Developer Tools for Eclipse, and Rational Developer for System z.



## Modern interfaces – Liberty Profile Web Container

- New Java web container is built on WebSphere Application Server Liberty profile technology:
  - Liberty is a lightweight, composable, ‘profile’ of WebSphere Application Server
    - Provides a fast and lightweight Java web container
  - Provides “off the shelf” Web-server capabilities (JSPs and Servlets)
  - Provides potential to re-use even more WebSphere technology in CICS.
  - JSP and Web servlets have direct, local, access to CICS data and resources.
  - Servlets can take advantage of existing CICS OSGi applications to provide a Dynamic Web front end.

## Notes ...

The WebSphere® Application Server Liberty profile is a fast and lightweight Java™ OSGi web container that runs in a JVM server. Using the Liberty profile, developers can use the rich features of the Java Servlet and JavaServer Pages (JSP) specifications to write modern web applications for CICS®.

The Liberty profile helps developers quickly and easily develop and test applications in a web server with a minimum of configuration and a quick start up time. Developers can easily change and add features to the web server as they develop an application, and the quick startup time means that changes can quickly be tested. The Liberty profile can be used in combination with a wide range of free tools that comply with Java EE standards.

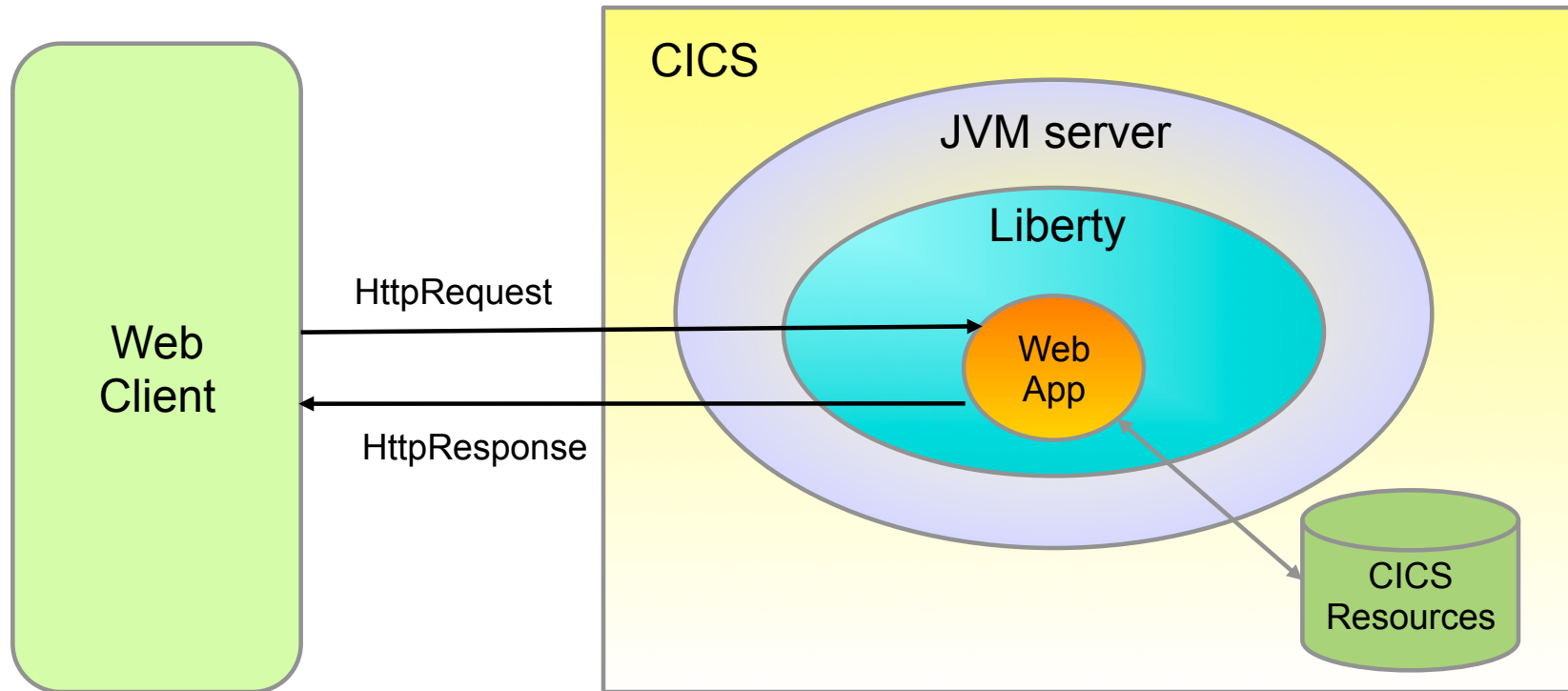
The Liberty profile is included with CICS to provide a runtime environment for Java web applications, such as servlets and JSP applications. The Liberty profile uses an OSGi framework and so fits in the existing JVM server architecture, combining the ability to develop modern web applications with a robust and scalable CICS JVM server environment.

Key benefits for running the Liberty profile include, but are not limited to, the following capabilities:

- Using Eclipse-based developer tools for broad tooling support, including WebSphere Application Server Developer Tools for Eclipse (WDT) and Rational® Developer for System z®.
- Modernizing presentation interfaces for CICS applications, from 3270 to web browser and RESTful clients.
- Packaging and deploying web applications from the CICS Explorer® SDK.
- Running the web presentation layer and business logic in the same JVM server.



## Modern interfaces – Liberty Profile WEB Container



- Liberty Profile runs in a JVMSERVER
  - Use sample JVMSERVER profile DFHWLP
- Web App developed and deployed using Eclipse IDE & CICS Explorer SDK

## Notes ...

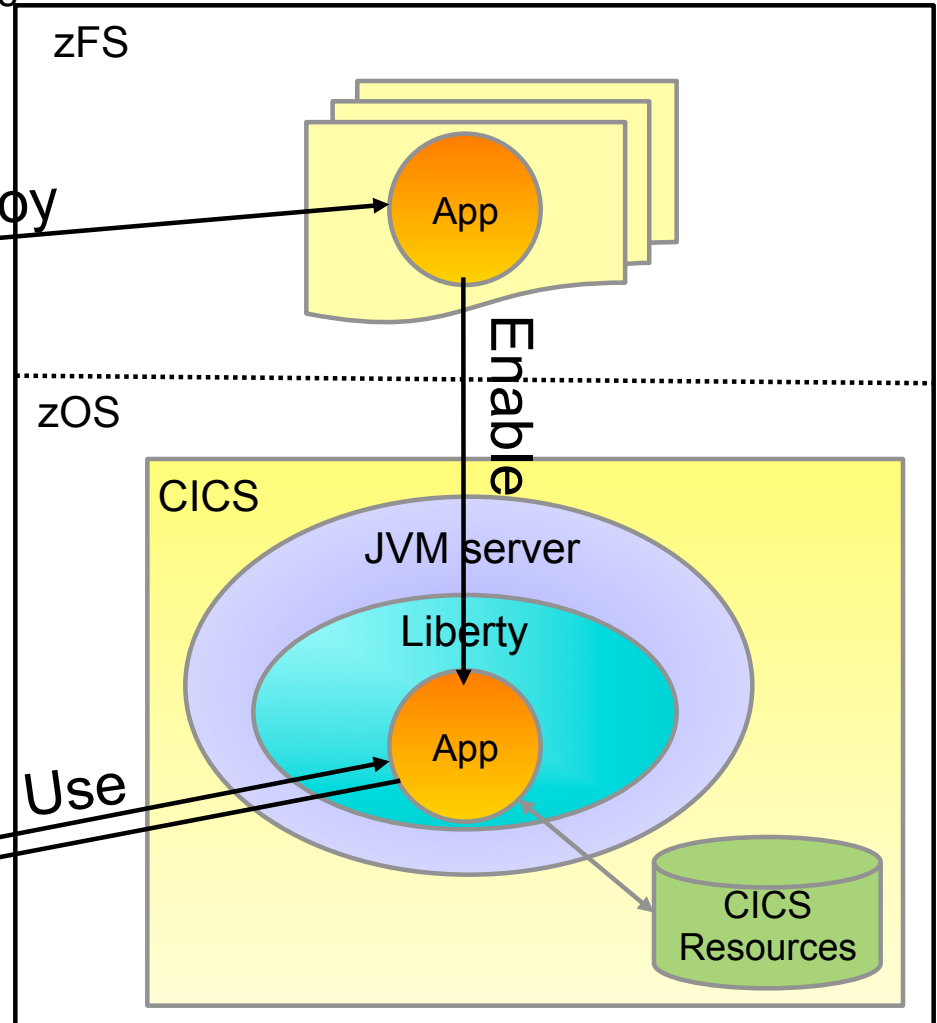
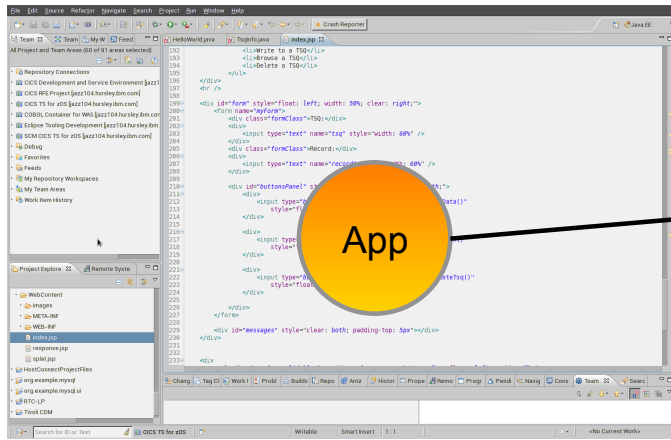
CICS provides a sample profile, DFHWLP, to be used to run a Liberty profile instance in a JVM server. Only minimal tailoring is required ie Uncomment the WLP\_SERVER\_HTTP\_PORT and choose a unique port number.





## Modern interfaces - Putting it all together

Eclipse with CICS Explorer SDK & Liberty Tools



## Notes ...

The CICS Explorer SDK contains examples to help you start developing servlets and JSP pages that can run in a Liberty profile instance in CICS. You can use the CICS bundle project that is supplied with each servlet example to deploy the application to a Liberty profile server that is running in CICS.

CICS loads and runs Java™ applications from zFS, so you must deploy your compiled applications to a directory in zFS that the CICS region can access.

The preferred method to deploy and install the applications is to use a CICS bundle project, because you can then use CICS facilities to manage those bundle project application resources. A CICS bundle project packages a set of plug-in projects, dynamic web projects, and CICS resources together that are logically deployed and managed as a single unit. You can use this method for simple web applications, such as the Hello World example, and more complex web applications, such as the other servlet examples.

An alternative method to deploy and install the applications is to use the Liberty deployment mechanism to deploy WAR files. You can export the dynamic web project as a WAR file and use FTP to copy the file to the drop-in directory of a Liberty profile server. You can use this method for simple web applications, such as the Hello World example.



## Modern interfaces - JCICS ExecutorService

- Thread.start equivalent (from Java concurrency package)
  - A standard Java pattern for dispatching runnable code to threads
- CICS provides “CICSExecutorService” - to create CICS capable threads
  - CICSExecutorService registered with OSGi registry, can be obtained and used by 'vendor' products and applications
- A convenience method provided called “CICSExecutorService.runAsCICS()”
  - Liberty requests an ExecutorService from the OSGi service registry. When running in CICS JVM server, it is given the CICSExecutorService which produces JCICS enabled threads for Liberty to run servlets on
- Each invocation (servlet request) has a CICS task and UOW environment with full capabilities

## Notes ...

CICS provides an implementation of the Java ExecutorService interface. This implementation creates threads that can use the JCICS API to access CICS services. When the JVM server starts up, it registers the ExecutorService as an OSGi service. Use this implementation instead of the Thread.start() method. The support is provided by the following new JCICS class:

### CICSExecutorService

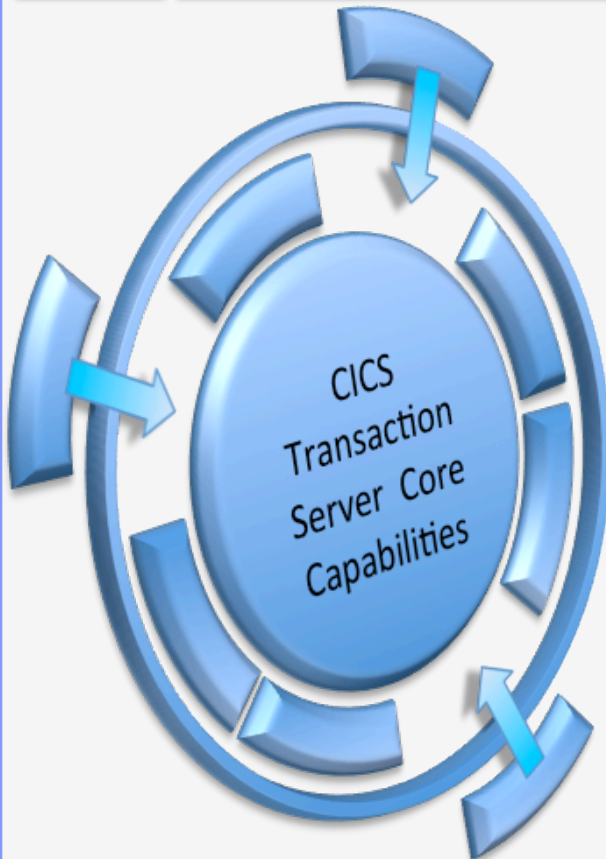
This class implements the `java.util.concurrent.ExecutorService` interface. It provides a static version of the execute method called `runAsCICS()`. The static method checks in the OSGi service registry for an instance of the `ExecutorService` for the application.

Because this class is registered as an implementation of the Java `ExecutorService` interface, any application that requests an `ExecutorService` is given the `CICSExecutorService` only when running in a JVM server. The application is decoupled from the implementation and can therefore run in any environment.

## Increasing service agility - Foundational enhancements



## Extending core CICS capabilities



### Enhanced in a number of areas...

- Enhancements to CICS event support, including "one-to-many" emission
- Support of greater-than-32KB messages across the CICS-WebSphere MQ Dynamic Program Link (DPL) bridge
- Enhancements to IPIC to add IMS support and improve integration and error recovery
- Reduced application storage needs with GET and PUT container enhancements
- Backup and restore capability for entire CICSplex System Manager (CICSplex SM) systems without manual overrides
- Automatic adjustment of the CICS clock for daylight saving time changes

## Notes ...

Foundational enhancements extend the core CICS capabilities in a number of areas. This slide summarises the key improvements which are described in more detail in the next few slides.



## Foundational enhancements

- **Multiple Event Processing Adaptors**
  - You can emit an event to several event consumers using the new event processing adapter sets.
  
- **CICS-WebSphere MQ DPL Bridge**
  - Supports a Channel/Container based interface
  - One request container, one response container
  - May be further enhanced in future to allow multiple containers
  
- **Support IPIC for IMS TM to CICS TS communications**
  - Async START RETRIEVE only
  - Requires IMS rnext. See Software announcement October 3rd 2012
  
- **PUT CONTAINER command has new APPEND option**
  - Enables containers to “grow” without having to delete/recreate
  
- **GET CONTAINER command has new BYTEOFFSET option**
  - Retrieve a section of container data beginning at offset

## Notes ...

Using the EP Adapter Set editor, you can now specify multiple EP adapters and configure an EP adapter set. You can then choose to emit events either to an EP adapter or an EP adapter set in the event binding editor.

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 DFHMQB3. 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 DFHMQB0 and pass and receive data by using the COMMAREA.

IMS next will support IMS TM to CICS communication using IPIC for async start retrieve type requests.

The APPEND option on PUT CONATINER specifies that the data passed to the container is appended to the existing data in the container. If this option is not stated, the existing data in the container is overwritten by the data passed to the container.

The BYTEOFFSET option on GET CONATINER specifies the offset in bytes where the data returned starts. For CHAR containers, the BYTEOFFSET value is used as an offset into the data in the requested codepage. If you use a codepage with multibyte characters, depending on the BYTEOFFSET value you specify, the data returned might have partial characters at the beginning, end, or both. In this situation, your application program must be able to handle and interpret the data returned. If the value specified is less than zero, zero is assumed.





## Foundational enhancements

- EYU9XDBT utility for CMAS and CICSplex definition import/export
  - Relationships between CMASes and CICSplex definitions retained
  - Restores the correct inherit relationships between RTA, Monitor, and Workload specifications and the CICS regions
  
- AUTORESETTIME=IMMEDIATE is now the default
  - CICS issues a PERFORM RESET command to synchronize the CICS time-of-day with the system time-of-day if, at the next task attach, the CICS time-of-day differs from the system time-of-day.

## Notes ...

EYU9XDBT, the CICSplex® SM data repository utility program, has enhanced functional and reporting capability.

EYU9XDBT enables you to export and import complete CICSplex SM data repository backups, at the level of a CMAS or a CICSplex context. The following enhancements complete this capability:

Relationships between CMASes and CICSplex definitions can now be exported from one CMAS and re-imported to a new CMAS, retaining the CPLXCMAS associations.

Relationships between RTA, Monitor, and Workload specifications and the CICS regions with which they are associated now have their creation modes retained when imported to a new CICSplex. Previously, INHERIT relationships between a WLM Specification (WLMSPEC), RTA Specification (RTASPEC), or Monitor Specification (MONSPEC) and a CICS region were converted to EXPLICIT relationships when they were imported to a new CICSplex. INHERIT relationships can only occur when you associate these specifications with a CICS system group (CSYSGRP). EYU9XDBT now ignores any link records in an import that specify an INHERIT relationship between these specifications and a CICS region, and automatically restores the correct INHERIT relationships when the link records for the parent CSYSGRP are imported.

EYU9XDBT also reports more summary data for each command processed. In addition to the existing Command Execution Summary and Data Repository Access Summary reports, EYU9XDBT now provides a command execution summary by resource type. For example, the new data might show that 2 CICSplex objects were defined, or 3 WLMSPEC records were imported.

CICS® can now handle time changes for daylight savings automatically, so that the need for operator intervention is removed.

The AUTORESETTIME system initialization parameter specifies the action CICS takes for automatic time changes. In CICS TS 4.2 and earlier releases, if this parameter was enabled, the CICS time-of-day was synchronized with the system time-of-day at the next local midnight.

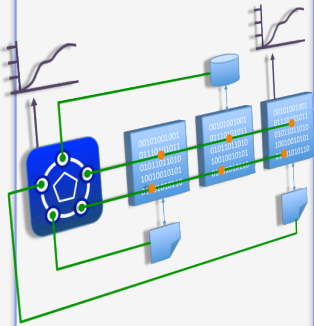
In CICS TS for z/OS®, Version 5.1 , the AUTORESETTIME system initialization parameter has a new option, IMMEDIATE. This option is the default. When this option is enabled, the time-of-day synchronization occurs at the next task attach.

## CICS TS 5.1 - Increasing Service Agility



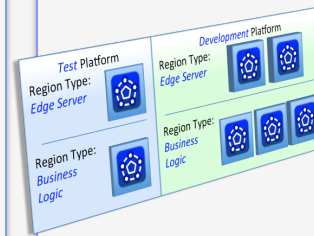
### First-class Applications

- Manage disparate resources as a single entity
- Rapidly move through the application lifecycle
- Automate dependency management
- Ensure rigorous yet flexible provisioning
- Measure entire application resource usage
- Dynamically manage applications with policies



### First-class Platforms

- Group new and existing regions as platforms
- Decoupling applications from the region topology.
- automatic resource deployment and validation
- De-provision resources when requested
- Deploy applications to regions within a platform
- Dynamically manage platforms with policies



### Modern Interfaces

- A production-ready web container
- Deploy lightweight Java servlets and JSPs
- Local access to CICS applications and data
- Roll-out of interface updates through OSGi
- Integration with applications and platforms
- Built on WebSphere Application Server Liberty profile for compatibility



### Foundational Enhancements

- CICS supports "one-to-many" event emission
- Greater-than-32KB across MQ (DPL) bridge
- Enhancements to IPIC add IMS support
- Reduced application storage needs with GET and PUT container
- Backup and restore entire CICSplex System Manager (CICSplex SM) systems
- Automatic adjustment of the CICS clock for daylight saving time changes



## Notes ...

This slide lists all the enhancements made in CICS TS 5.1 to address the challenge of Increasing service agility, grouped under the four key areas.



## CICS Explorer 5.1 and CICS Explorer SDK 5.1

- Significant new capability to support all relevant aspects of CICS TS V5.1
  - Applications, including dependencies and entry points
  - Platforms
  - Policies
  - Event bindings and adapter sets
  - Web application samples, development, and deployment
- New z/OS Management Facility (z/OSMF) support
- Productivity and usability improvements, for example:
  - Create new z/OS UNIX files, data sets
  - Perform a system dump and SSL rebuilds
  - Create a new resource from an existing definition
  - Copy and paste resource data to a spreadsheet

## Notes ...


CICS Explorer 5.1 contains significant enhancements to the CICS Explorer and the CICS Explorer SDK over and above the release of CICS Explorer for CICS TS 4.2 and the interim CICS Explorer 1.11

CICS Explorer and the CICS Explorer SDK provide support for the new concepts in CICS TS 5.1, notable, applications, platforms and policies plus support for Liberty Profile.

## Summary: The new CICS TS V5.1 release

### 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



© Copyright IBM Corporation 2012. All rights reserved. The information contained in these materials is provided for informational purposes only, and 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, these materials. Nothing contained in these materials 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 these materials 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 these materials 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. IBM, the IBM logo, Rational, the Rational logo, Telelogic, the Telelogic logo, and other IBM products and services are trademarks of the International Business Machines Corporation, in the United States, other countries or both. Other company, product, or service names may be trademarks or service marks of others.