IBM VSE/ESA
CICS Transaction Server
Performance
Considerations

Dan Janda
VM/VSE System Center
Endicott, New York
janda at us.ibm.com

Wolfgang Kraemer
VSE Product Mgmnt
71032-15 Boeblingen
wkraemer at de.ibm.com

Tech Conf Update 2000-03-31

RETURN TO INDEX

Contents

Contents

E. CICS TS New Operational Aspects

F. CICS TS Familiar Facilities

G. CICS Intercommunication

H. CICS TS System Initialization Table

Contents

I. CICS TS Measurement Results/Systems

J. App. M: VSE/ESA 2.4 Perf. Enhancements

Notes

Note

All information contained in this document has been collected and is presented based on the current status.

It is the responsibility of any user of this VSE/ESA V2 document

- to use the latest update of this document
- to use this performance data appropriately

This document is unclassified and suited for VSE customers. It is part of the package VE21PERF which resides on an IBM disk, called IBMVSE.

Access to the VE21PERF package is available for IBM representatives, just by typing under CMS

TOOLS SENDTO BOEVM3 VMTOOLS IBMVSE GET VE21PERF PACKAGE

These VSE performance documents are also available from INTERNET via the VSE/ESA home page

http://www.ibm.com/s390/vse/

or directly via FTP links

http://www.ibm.com/s390/vse/vsehtmls/s390ftp.htm

Starting with the VSE/ESA 2.4 documentation, these documents are also available on the VSE/ESA CD-ROM kit SK2T-0060.

The following documents are available in Adobe Reader format (.PDF):

'IBM VSE/ESA 1.3/1.4 Performance Considerations'
'IBM VSE/ESA V2 Performance Considerations'
'IBM VSE/ESA Turbo Dispatcher Performance'
'IBM VSE/ESA I/O Subsystem Performance Considerations'
'IBM VSE/ESA VM Guest Performance Considerations'
'IBM VSE/ESA Hints for Performance Activities'
'IBM VSE/ESA TCP/IP Performance Considerations'
'IBM DFSORT/VSE Performance Considerations'
'IBM VSE/ESA CICS Transaction Server Performance' (this document)
'IBM VSE/ESA V2.5 Performance Considerations' (to come)

Trademarks

The following terms included in this paper are trademarks of IBM:
ES/9000 ESA/390 System/390 SQL/DS PR/SM
VM/ESA VSE/ESA ESCON ECKD n-ways
CICS Transaction Server DB2

EXPLORE/VSE is a trademark of Computer Associates
TMON/VSE is a trademark of Landmark Corporation

Notes ...

Disclaimer

This document has not been subjected to any formal review or testing procedures and has not been checked in all details for technical accuracy. Results must be individually evaluated for applicability to a particular installation.

Any performance data contained in this publication was obtained in a controlled environment based on the use of specific data and is presented only to illustrate techniques and procedures to assist to understand IBM products better.

The results which may be obtained in other operating environments may vary significantly. Users of this document should verify the applicability of this data in their specific environment.

The above disclaimer is required since not all dependencies can be described in this type of document.

Acknowledgements

Thanks to all who contributed directly or indirectly, be it by measurements, suggestions or in other ways.

Special thanks is expressed to

- Hanns-J. Uhl for VSE system support and measurements
- Hans J. Ebert from the VM/VSE Center Munich, Germany
  for CICS support and tons of experience
  with customer production CICSs
- Ingolf Salm the designer of the Turbo Dispatcher
- Stanley Jones from the VM/VSE System Center Endicott, NY
  for technical and 'mental' support
- Neville Brailsford from CICS Development Hursley, UK
- Andy Wharmby from CICS Development Hursley, UK

All mistakes and inaccuracies in this document are owned by us.

Please, as in the past, contact us if you have

- suggestions or questions regarding this document

- questions on VSE/ESA performance, not covered in any of the VSE/ESA performance documents

Wolfgang Kraemer, IBM VSE Development, Boeblingen Lab, Germany
Daniel F. Janda, IBM VM/VSE Support Center, Endicott, New York

Notes ...

Remark to this Version of the Document

VSE/ESA 2.4 together with CICS TS 1.1 was made available 12/98 in a LIMITED AVAILABILITY fashion.

General Availability was 06/99.

This document will be enhanced and updated due to

- numerous new CICS functions, together with numerous new performance aspects
- the fact that still performance relevant changes are being included into the product
- still some ongoing performance measurements.

What has been added/changed?

Deltas as compared to earlier versions

Editorial changes are done throughout the document without special notice

„ Updates as of 99-03-29
- Minor refinements all over
- SIT Overrides for DSAs and EDSAs
- AUXTrace setup for performance purposes
- Journaling documentation improved
- Better setup of VSE/ESA 2.4 Migration Summary

„ Updates as of 99-06-25
- Changes in Migration Summary figures (GA data)
- DL/I 1.11 SVA=NO generation parameter
- Included some CICS TS SIT parameter variation results

„ Updates as of 99-10-25
- Added CICS VSAM Recovery 1.2

„ Updates as of 2000-xx-xx
- Miscellaneous editorial changes

Related Documents

This document deals with the specific performance aspects for the CICS Transaction Server for VSE/ESA 1.1

For general VSE/ESA V1 and V2 performance, refer to the documents

'IBM VSE/ESA 1.1/1.2 Performance Considerations'
'IBM VSE/ESA 1.3/1.4 Performance Considerations'
'IBM VSE/ESA V2 Performance Considerations'
'IBM VSE/ESA Turbo Dispatcher Performance'
'IBM VSE/ESA I/O Subsystem Perf. Considerations'
'IBM VSE/ESA VM Guest Perf. Considerations'
'IBM VSE/ESA Hints for Performance Activities'
'IBM VSE/ESA TCP/IP Performance Considerations'
'IBM DFSORT/VSE Performance Considerations'
'IBM VSE/ESA V2.5 Performance Considerations' (to come)

The VSE/ESA V2 base document is available since the 2.1 General Availability 04/95, it has been updated many times, and now contains also VSE/ESA 2.2, and 2.3.
VSE/ESA 2.4 information was appended in the CICS TS document.

These documents are also available to any IBM person, as part of the VExyPERF PACKAGEs on the IBMVSE tools disk:

TOOLS SENDTO BOEVM3 VMTOOLS IBMVSE GET VExyPERF PACKAGE
(xy = 12, 13 or 21).

Please contact your IBM representative to obtain an update of this document, if your copy is more than 6 months old and you do not have Internet access.

All documents are also available from INTERNET via the VSE/ESA homepage

http://www.ibm.com/s390/vse/

or directly via FTP links
http://www.ibm.com/s390/vse/vsehtmls/s390ftp.htm

in Adobe Reader format, as file VE13PERF.PDF, VEVMPERF.PDF, VE21PERF.PDF, VEIOPERF.PDF, VEPERACT.PDF, VETCPPER.PDF, VESORTP.PDF, VE21TDP.PDF, VECICSTS.PDF (this document) and VE25PERF.PDF.

Glossary

AOR	Application-owning region
ATI	Automatic Transaction Initiation. Initiation of a CICS txn by an internally-generated request e.g. the issue of an EXEC CICS START command.
BSM	Basic Security Manager. Supplied as part of VSE/ESA 2.4, RACROUTE-compliant.
CSD	CICS System Defintition Data Set. A VSAM KSDS file with RDO records.
DMF	Data Management Facility. A new item of CICS TS, running in a separate partition. Gets and handles all statistics and monitoring data.
DIM	Data in Memory. A concept to store as much data as practical in processor storage.
DSW	Data Systems Workload, used to assess CICS performance
ESM	External Security Manager. Any RACROUTE-compliant program that provides security checking functions.
FOR	File-owning region
FS	Function Shipping
GCD	CICS Global Catalog Data Set. A VSAM KSDS file to save all resource definitions that are installed.
ISC	CICS InterSystem Communication. Provides communication between CICS partitions in different VSE systems for TR and FS, or between CICS subsystems running in other environments.
ITR	Internal Throughput Rate A measure for processor and/or S/W effectiveness: Number of transactions or batch jobs per CPU-second. On n-ways it is per n CPU-seconds, thus ITR higher.

Glossary ...

ITRR Internal Throughput Rate Ratio.
Ratio between the ITR of a processor and S/W environment and the ITR of a (base) processor and a (base) S/W setup. It simply can mean a power ratio between processors, or a (inverse) ratio of CPU-times between 2 releases.

LCD CICS Local Catalog Data Set.
A VSAM KSDS file where CICS saves additional information between CICS runs.

LSPR Large System Performance Reference.
IBMs method to characterize relative processor speed. Based on measurements.

MP, n-way These terms are used here interchangeably.
Any processor system with more than 1 engines ('CEC's), sharing processor storage and I/O subsystem/channels.

MPS Multi Partition Support (for DL/I).
A DL/I data base under CICS can also be accessed by DL/I applications in other (batch) partitions

MRO CICS Multiple Region Option.
Provides communication between CICS partitions using Transaction Routing (TR) or Function Shipping (FS).

NP Non-Parallel code that cannot run in parallel on more than 1 processor.

PAM Page Allocation Map

PB Partition Balancing, a VSE function.
With Turbo Dispatcher you can also specify 'VSE relative processing weights'.

RCF CICS Report Controller Feature.
A CICS feature that provides distributed control over report printing and printing devices.

RDO Resource Definition Online.
The recommended method of defining resources to CICS.

Glossary ...

Region here used interchangeably for VSE 'partition'.

RSD CICS Restart Data Set.
A VSAM KSDS file with backout information used only during emergency restart.

SSP Subsystem Storage Protection.
A CICS TS function that protects CICS code and control blocks from accidental overwrite by user applications.

TD VSE/ESA Turbo Dispatcher for support of n-ways.
Also usually used for CICS Transient Data

TempStor CICS Temporary Storage
(used instead of TS, to avoid confusion).

TOR Terminal-owning region

TR Transaction Routing.

TranData Used in this document for CICS Transient Data

TS Transaction Server.
NOTE: Usually used for CICS Temporary Storage, called 'TempStor' here, to avoid confusion.

txn Used in this document for 'transaction'

VSCR Virtual Storage Constraint Relief.
Any means to free virtual storage below the 16M line.
Allows for more 24-bit dependent programs and area
and thus results in a higher partition capacity
plus use of DIM.

XRF Extended Recovery Feature

Further References

The following are references for further (also performance related) information in the context of VSE/ESA V2.4 and CICS TS:

CICS TS for VSE/ESA 1.1 Library

- Performance Guide SC33-1667-00
- System Definition Gude SC33-1651-00
- Customization Guide SC33-1652-00
- Resource Definition Guide SC33-1653-00
- Operations and Utilities Guide SC33-1654-00
- CICS Supplied Transactions SC33-1654-00
- Shared Data Tables Guide SC33-1668-00
- Intercommunication Guide SC33-1665-00
- Migration Guide SC33-1666-00

Conference Contributions

Ù CICS/VSE Shutdown Performance: Using Shutdown Statistics by Dan Janda, IBM US, Endicott VM/VSE Tech Conf., 05/96

Ù VSE/VSAM and CICS/VSAM Performance, by Horst Sinram and Dan Janda IBM Germany, Boeblingen and IBM US, Endicott VM/VSE Tech Conf., La Hulpe, 09/96

Ù CICS: Tips and Techniques (TNT), by Stanley Jones IBM US, Endicott. VM/VSE Tech Conf Reno, Nevada 05/98

Ù An Overview of CICS TS for VSE/ESA, by Andy Wharmby, IBM UK, Hursley. VM/VSE Tech Conf., Orlando, 05/99

Ù Application Programming on the new CICS, by Neville Brailsford, IBM UK, Hursley. VM/VSE Tech Conf., Reno, 05/98

Further References (cont'd)

Redbooks

The following IBM Redbooks are also available:

'Migration to VSE/ESA 2.4 and CICS TS ... ', SG24-5595-00

'Implementation of VSE/ESA 2.4 and CICS TS ...', SG24-5624-00

More Info

- New Articles Related to CICS TS:

  VSE/ESA Software Newsletter, 3rd/4th Quarter 1999,
  G225-4508-19, 12/99

- Accessible e.g. via the CICS homepage

  http://www.software.ibm.com/ts/cics/

PART A.

CICS TS Overview

- Overview
- Basic Concepts
- Performance Relevant Items

CICS TS 1.1 Overview (1/2)

Ù Functional level as of CICS/ESA for MVS 4.1 + PTFs

„ CICS Command-level (API) only

Much easier to use and less error prone than Macro-level.

„ More RDO

Now e.g. also for VSAM files (incl. LSR pools, Data Tables) and for remote files

„ More dynamic space management

Individual DSA sizes are dynamically adjusted

„ Shared Data Tables

Replace the Data Tables from CICS/VSE 2.2/2.3, with important advantages over the old support

„ CICS Subsystem Storage Protection (SSP)

Separate the CICS code from application code and data:
Prevent CICS code and control blocks from being overwritten accidently by user applications (on ESA/390 processors).

Use of separate storage keys by CICS modules and applications.
Function differs from 'Transaction Isolation' (user vs user).

„ Support of VTAM Persistent Sessions

VTAM sessions maintained across CICS outage, w/o need to rebind, an alternative to XRF.
The Inter-Enterprise variant of VTAM is required

Cont'd on next page

CICS TS 1.1 Overview (2/2)

Ù CICS Report Controller Feature (RCF)

Taken over from CICS/VSE, some functional enhancements

Ù Support of DL/I VSE 1.11

Ù CICS/DDM 1.1

CICS DDM (Distributed Data Management) allows (function-ship like) access from OS/400 to VSAM data.

Now a no-charge optional feature

Ù Support of CICS/VR 1.2

New release for CICS-VSAM Recovery

Ù REXX for CICS

To be available soon

Ù CICS Web Support

Allows direct communication from a Web browser.

Taken over from CICS TS 1.3 for OS/390. Needs TCP/IP for VSE/ESA (will be described in the VSE/ESA 2.5 document).

To be available soon

CICS TS 1.1 Overview ...

Major CICS TS Benefits

- **Increased Availability**
- **Improved System Management**
- **Improved Restart and Recovery Mechanisms**
- **Improved Security Services**
- **Improved Programming Support**
- **Improved Performance/Capacity**

Dropped Support vs CICS/VSE 2.3

- Macro-Level API
- BTAM
- Direct control block access (CSA, TCA, etc.)
- Support of some old language translators (RPG, C/370, DOS PL/I)
- Internal Security (but basic functions provided by VSE/ESA)

Dropped functions still supported in a CICS/VSE partition, potentially connected via MRO to CICS TS.

- **CICS Coexistence Environment**

Ù Restructured CICS code

í Domain Concept

Functionally isolated parts of CICS that can communicate with the rest of CICS by means of strictly-defined standard interfaces.

Ù MVS based code

í OS/390 Emulation Mode

MVS macros were kept in CICS, and are interpreted by the VSE/ESA 2.4 'Family API'.

This 'OS/390 Emulation Mode' is enabled via EXEC xxxx,OS390.

í Cross-Memory Communication

Implemented via Space Switching Program Call (PC-ss).
Exploited by CICS Shared Data Tables and DMF

Ù Moving CICS code and control blocks above the line

í VSCR

Majority of the CICS nucleus and control blocks

Some code must remain below the line (they use services that do not support 31-bit addressing, about 300K):
Some code for TranData, Journaling, Txn dumps, AUXtrace

Refer to foil 'CICS TS Areas Required Below the Line'

CICS TS Performance Relevant Line Items

Major Performance Relevant Items in CICS TS 1.1

- ### CICS VSCR

  Significant relief to increase CICS capacity and to even more DIM

- ### 31-bit storage now fully managed by CICS

  'DSA above the line' with many subpools

- ### Shared Data Tables

  Adds generic key usage and Browse.
  Avoid FS overhead for 'remote' READs (incl. Browse).

- ### Separate VSAM LSR (sub)pools for index and data

  Index and data (which have different access patterns) may reside in separate pools (and thus subpools), even when CI-size is same.

  In total, more LSR (sub)pools possible, may be smaller at same degree of DIM exploitation.

- ### Dynamic Transaction Routing

  A user-replaceable module can now decide which target CICS to be used for load balancing, or other reasons

- ### Support of DL/I VSE 1.11

  #### Multiple DL/I MPS support can be exploited

  (VSCR functions were also available as PTFs to DL/I DOS/VS 1.10.
  Refer to 'IBM VSE/ESA V2 Performance Considerations').

  #### More than 4 GB for HD data bases

  DL/I data bases can be allocated on multiple VSAM ESDSs. Up to 5 datasets are used internally, transparent to user applications.

- No VSAM or VTAM subtasking exploited in VSE (and by Turbo Dispatcher) to extend the n-way exploitation further

CICS VSAM Recovery V1.2 (CICSVR 1.2)

Program Number is 5686-011. Ported from CICSVR/MVS 2.3.
Announced 99-11-30, available since 12/99.

Additional functional enhancement for CICS/VSE Automatic Journal Archive in 04/2000.

CICSVR 1.2 Enhancements vs CICSVR 1.1

Support of CICS Transaction Server

Besides CICS/VSE 2.3

Support of Transaction Backout

CICS TS only

Enhanced Forward Recovery for VSAM ESDS

ESDS inserts that have been marked by CICS for deletion are handled correctly by CICSVR (no exits required)

Support of Automatic Journal Archive

In CICS TS via a new CICS TS function.

In CICS/VSE provided via PTF for APAR HC58567 (PTF available about 04/2000):

- CICS/VSE user-replaceable programs provided:
  DFHXJCO and DFHXJCC
- Skeleton jobs for all required activities
  (including an archive tracking file)

Refer to the CICSVR User's Guide and Reference (Chapter 7)

Functions of CICSVR 1.2

Automate your VSAM recovery procedures:

Automatically archive CICS system and user journals

Set up CICS to use CICSVR to automatically archive CICS system and user journals when the journal is closed

In case of a recovery situation ...

Restore the VSAM data set from Backup tape

Perform Forward Recovery with CICSVR

CICS journal records (with after-images) are used

In case of CICS TS online backout failure ...

Perform Backout with CICSVR

CICS system log records (with before-images) are used.

CICS VSAM Recovery VSE/ESA V1.2 ...

CICS VSAM Recovery V1.2 (cont'd)

- **Types of situations of VSAM 'Data Damage'**

  User errors

  Unintentional DELETEs, changes...

  Head crash and unrecoverable I/O errors

  Need for CICSVR reduced(?) by RAID (-1, -5, -6) I/O subsystems

  Corrupted data

  Disaster (fire etc.)

- **Supported VSAM file types**

  All, except VSAM VRDS

  ESDS *)
  KSDS *)
  RRDS

  *) Base cluster and associated paths (alternate index)

- **Limitations. Naturally, CICSVR cannot ...**

  forward recover a VSAM file that does not have an associated journal containing after-images

  backout a VSAM file that does not have an associated system log containing before-images

  recover Batch (non-CICS) updates of a VSAM file

PART B.

General CICS Performance Aspects

- Performance Design
- Transaction Types
- Performance Constraints
- Performance Trouble Shoot
- Load Balancing
- CPU Constraints
- More Hints and Tips
- Too Frequent Program Loads

Virtual Storage Constraint Relief (VSCR) is a pre-req for DIM

Data in Memory (DIM)

- **DIM is a cheap means to improve performance**

  - Reduce physical I/O operations
  - Save CPU-time

  - **Improved response times and higher CICS capacity**

- **If done in a reasonable way, the cost of additional real storage is really worth while**

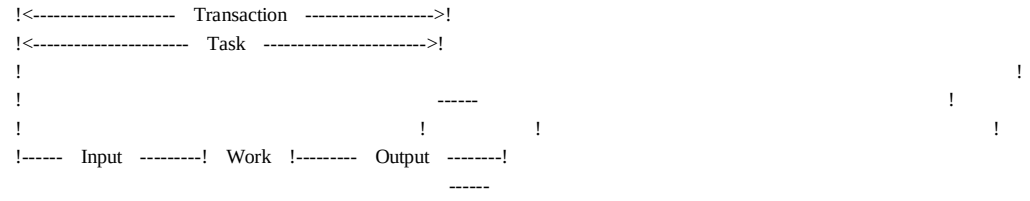
- **In no case you should 'overdim' and get page faults instead**

Refer to the detailed discussion of DIM in

'IBM VSE/ESA V1.3/1.4 Performance Considerations'

CICS Transaction Types and Tasks

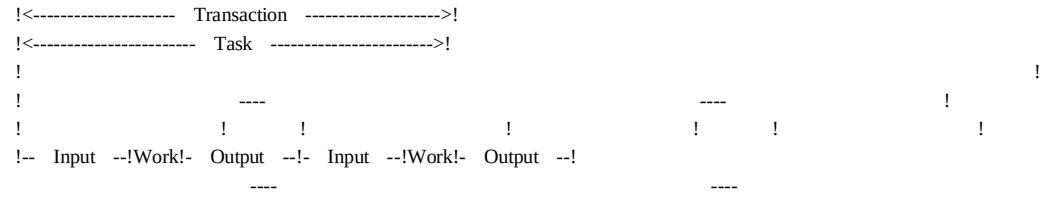
Ù Nonconversational Transactions (Txns)



- A CICS txn is identical to a CICS task and consists of 1 'message pair' ('txn') in performance terms

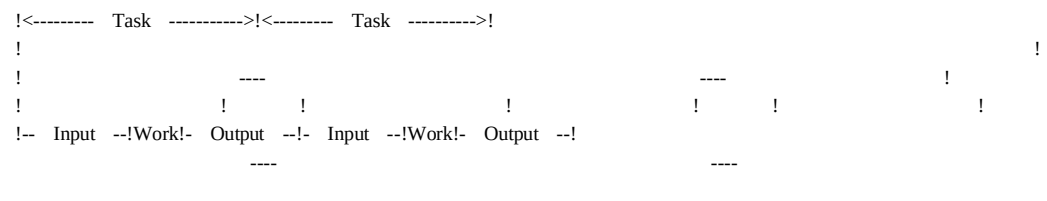
- Txn is initiated at first input, terminated after (last) output.
  CICS storage released. Data required later is written to TempStor
  or to other CICS functions (e.g. Commarea).

Ù Conversational Txns



- Transaction and tasks: as above
- An EXEC CICS CONVERSE may be used
- Resources are held during terminal thinktime
  (e.g. TCA w/ TWA, TIOA, Pgm and GETMAIN storage, file records...)

Ù Pseudo-Conversational Txns (preferred)



- 1 CICS transaction consists of several tasks
- Resources are NOT held during terminal thinktime, thus some additional initialization/termination overhead
- May need a specialized or own performance monitor, if running many tasks with a single txn-ID, e.g. SAP R/2

CICS Performance Dependencies

CICS performance depends on various factors
- **those being also relevant for any batch type of load**
- **CICS or Online specific ones**

- **General factors**
  - **Available CPU-power and partition priority**
  - **Available real (=central) storage**
  - **Available total virtual storage for the partition**
  - **I/O subsystem speed/load**

- **Workload/Application characteristics**
  - **Application demand for CPU, I/O, storage**
  - **Usage of CICS functions and resources and associated resources in other partitions**
    (e.g. DB2 for VM/VSE)

    ... in conjunction with system/CICS setup

- **CICS specific factors**
  - **Higher sensitivity to paging**
  - **Short-On-Storage conditions**
  - **CICS internal resource availability**
    - **task constraints, string constraints**
    - **any specific constraint in a CICS function**
  - **VTAM and/or TCP/IP setup**
  - **Terminal attachment and network**

CICS Performance Dependencies (cont'd)

Note that more than 1 reason may exist for having bad response times and/or low transaction throughput (actual CICS capacity).

Also, e.g., having an I/O bottleneck may increase or even cause a bottleneck in CICS storage areas because of high response times and larger resource hold times.

- **Actual CICS Online throughput (txn/sec) is determined by the**
  - **resulting average response time (RT)**
  - **number of active terminals**
  - **average user thinktime**
- **Any system or CICS bottleneck directly will be reflected in the RT**
- **If no system or CICS bottleneck exists ...**
  - **RTs are optimal, and**
  - **total throughput (txn/sec) is determined nearly exclusively from outside the VSE system**

CICS Performance Trouble Shoot

Principal Sequence (any type of problem)

1. **Find out major cause**
2. **Apply corrective action**
3. **Review the resulting changes**

Determine symptoms for perf. problem

Poor response time ...

at all loads for all txns

Major checks	Potential Action/Solution
- Paging?	Reduce working set/Provide more storage
- High CPU utiliz.?	Check startup options, traces, ... to reduce pathlength
- Low CPU utiliz.?	Check available CPU-power for CICS Check for bottlenecks (I/O, CICS ...)
- High utiliz. of major resources ?	Check frequency of functions used and setup of the function
- Low utiliz. of major resources ?	Check for application errors and artificial bottlenecks

at medium and high loads

Major checks	Potential Action/Solution
- See above ...	
- Limits reached?	Check for task/strno/buffer availability
- File I/O	Check msec/IO and I/O rates
- TP bottleneck?	Check for VTAM buffer expansions

CICS Performance Trouble Shoot ...

Poor response time ... (cont'd)

for certain txns only

Major checks	Potential Action/Solution
- Type of txns? Check	which type of txns suffer, and what they use/have in common MRO? I/O? CICS fctn? Data base? Amount of bytes transferred?

for certain terminals only

Major checks	Potential Action/Solution
- Type of terminal? Check	terminal definition
- Line utiliz.? Improve	capacity of that part of network

for all txns, but only when certain txns are active

Major checks	Potential Action/Solution
- Find contented resource(s)	Avoid monopolization

More Info

For more info regarding performance monitoring/troubleshoot/tuning, refer e.g. to

'IBM VSE/ESA Hints for Performance Activities'

and

'CICS TS for VSE/ESA Performance Guide', Chapter 8

CICS Load Balancing

Balancing of a CICS load means to control (balance) the processing of different transactions.

Usually, even when your VSE system is OK overall, preferring specific txns is (mostly) at cost of other ones.

CICS TS Means for Balancing

Ù CICS Txn Processing Priority

Transaction processing priority =

Txn priority + Terminal priority + Operator priority

Values can be changed via the master transaction

CEMT SET TXN!TERM PRIO(value)

Default Txn priority in CICS/VSE and TS is 1, so specific txns can be set lower (i.e. 0, e.g. for browsing tasks).

Í We recommend setting defaults to about 40 to 50.

Í Do NOT set individual values too high, in order to stay well below say 200 (i.e. to still keep flexibility).

The priority issue becomes more important nowadays,
- when DIM leaves less and less I/O waits
- where I/O subsystems provide very fast I/O response times

Ù Txn Priority

Reasons for High Priority	Reasons for Low Priority
- Minimize resource contention	- Lengthy Browsing
- Task with need for fast response (Data entry, time critical apps)	- Task w/o user interaction
	- CPU consuming background

CICS Load Balancing ...

Ù Dynamic Changes in Priority

- Priority Aging (CICS TS only)

Gives low-priority txns an increasing chance to finish in a system which is dominated by high-priority transactions: after PRTYAGE msec in the ready queue, task priority is incremented.

- TRANCLASS Limits (CICS TS only)

Txns belonging to a TRANCLASS(DFHTCLxx) are subject to scheduling constraints

- Change Txn Priority in Application (all CICSs)

In very specific cases, a CICS application program may be coded such that it shows a 'friendly behavior':
e.g. use

- EXEC CICS DELAY to avoid extreme monopolization
- EXEC CICS CHANGE PRIORITY when performance characteristic changes

Other Means

Ù Impact of specific tasks on other txns can be reduced (besides lower priority or other artificial constraints)

- use a separate LSR pool for BROWSE txns (or only subpool, if CI-size is unique)

For general VSE/ESA Workload Balancing, refer to the chapter in 'VSE/ESA 1.3/1.4 Performance Considerations'

CICS and CPU Constraints

Definition of 'CPU Constrained'

A CICS partition is ...

	Single processor util.
'CPU constrained'	>75%
'Totally CPU constrained' >90%	
- Values include any utilization of higher priority partitions and may refer to e.g. 15 min averages	

Important in Case of CPU Constraints

Ù Knowledge on how the problem evolved

The answer to the following questions can help a lot:

- Usual growth of #txns (higher business volume) ?
- More effective work done per txn (e.g. more items per order)?
- Any change of parameters or setup ?
- Any change in a CICS application or of vendor program release?

Ù Tuning Status of the CICS partition

- DIM already exploited via VSAM (NSR/LSR) buffers?
- Data Tables used?
- CICS Internal Tracing required?

Ù Collect workload and tuning status information

Keep for potential later use

BUT: Monitor the CICS partition BEFORE your txn volume grows to 90% CPU utilization

Otherwise it's very difficult to monitor (CPU overhead)

CICS and CPU Constraints ...

Means to Overcome CICS CPU constraints

1. Make sure CICS partition gets 'highest' dispatching priority

e.g. even higher than VSE/POWER (in case of high POWER activity)

2. Reduce CPU requirement by CICS tuning

- SIT parameters
- File definitions and access (for Data In Memory)
- Setup of MRO environment
- Design of applications
- ...

3. Determine CPU consuming txns and prevent them from monopolizing resources

4. Check the CPU-time overhead/impact of your CICS Monitor (and/or of your VSE System monitor)

Use e.g. QUERY TD CPU-times which include Monitor overhead time

Select 'better' monitoring options for lower overhead

5. Split CICS workload across multiple CICS partitions and use >1 engines

In order to avoid too much FS overhead, do an 'intelligent' split into AOR and FOR.

6. Migrate to a processor with more power per engine

More CICS Hints and Tips

- **You may move CICS/ICCF away from CICS production partition**

- **Think of using VSE Dynamic Partitions for CICSs when possible**

  - **No overhead when started**
  - **Save some SVA-24 space**

    Dynamic Space GETVIS is used for specific requests (SPACE=YES).
    May save about 100K per partition.

- **Application programming MUST NOT disregard common sense regarding performance design**

- **Before putting a new application into production ...**

  - **Compile the programs with the OPTIMIZE option.**

    This can save a lot of CPU-time (up to 30%).

    Is much simpler with COBOL/VSE than with DOS/VS COBOL

  - **Monitor the programs for system requirements**

    - CPU-time and # I/Os expected per transaction
    - estimated txn/sec

    Will the additional load fit in the remaining capacity?

    What will the effect be on other CICS and Batch workload?

CICS Checks for Frequent Program Loads

In case of frequent CICS program loads (observed from the CICS Program Statistics), you may proceed as follows:

- **Did program compressions or SOS situations occur?**
- **Check available private space via MAP and GETVIS Fx**

  Make sure you did not lose a 1M segment e.g. by a tiny increase in shared space-24 requirement
  (Partition then starts 1M higher)

- **Check available DSA size(s)**

CICS/VSE 2.3	CICS TS
- Size of the DSA	- Size of the DSA and of EDSA vs maximum possible size (DSALIMIT, EDSALIMIT).
- Check of PAM is not easy	- Size of the remaining space in the pertinent subpools

- **Check definitions of CICS programs**

  - You may have too many (seldom used) RES=YES programs
  - You may have unnecessarily defined programs with RELOAD=YES via RDO (GETVIS creep?)
  - A CEMT Set Program abcd NEwcopy may have been issued

- **Use 31-bit programs to get VSCR**
- **You may use a vendor program (X-ABOVE or equiv.) to move 24-bit applications above the line**

  presumably at cost of some CPU-time.
  (until you have moved to a new compiler and environment)

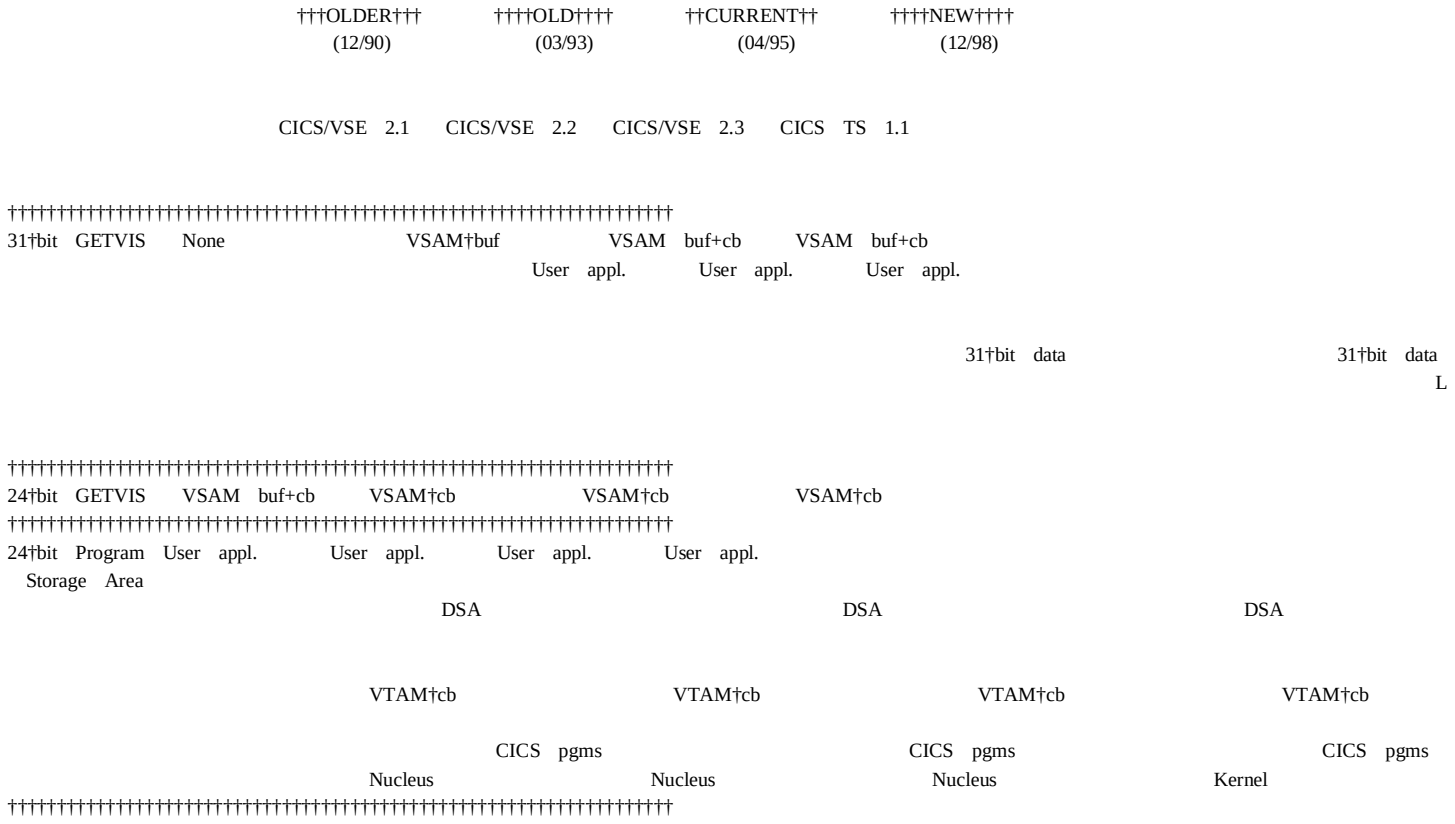
PART C.

CICS TS Virtual Storage Management

- Virtual Storage Evolution
- CICS/VSE 2.3 Storage Layout
- CICS TS Storage Layout
- Dynamic DSA Management
- Handling of DSA Shortage
- Kernel Storage, Message Table
- Areas Required Below the Line
- VS Recommendations

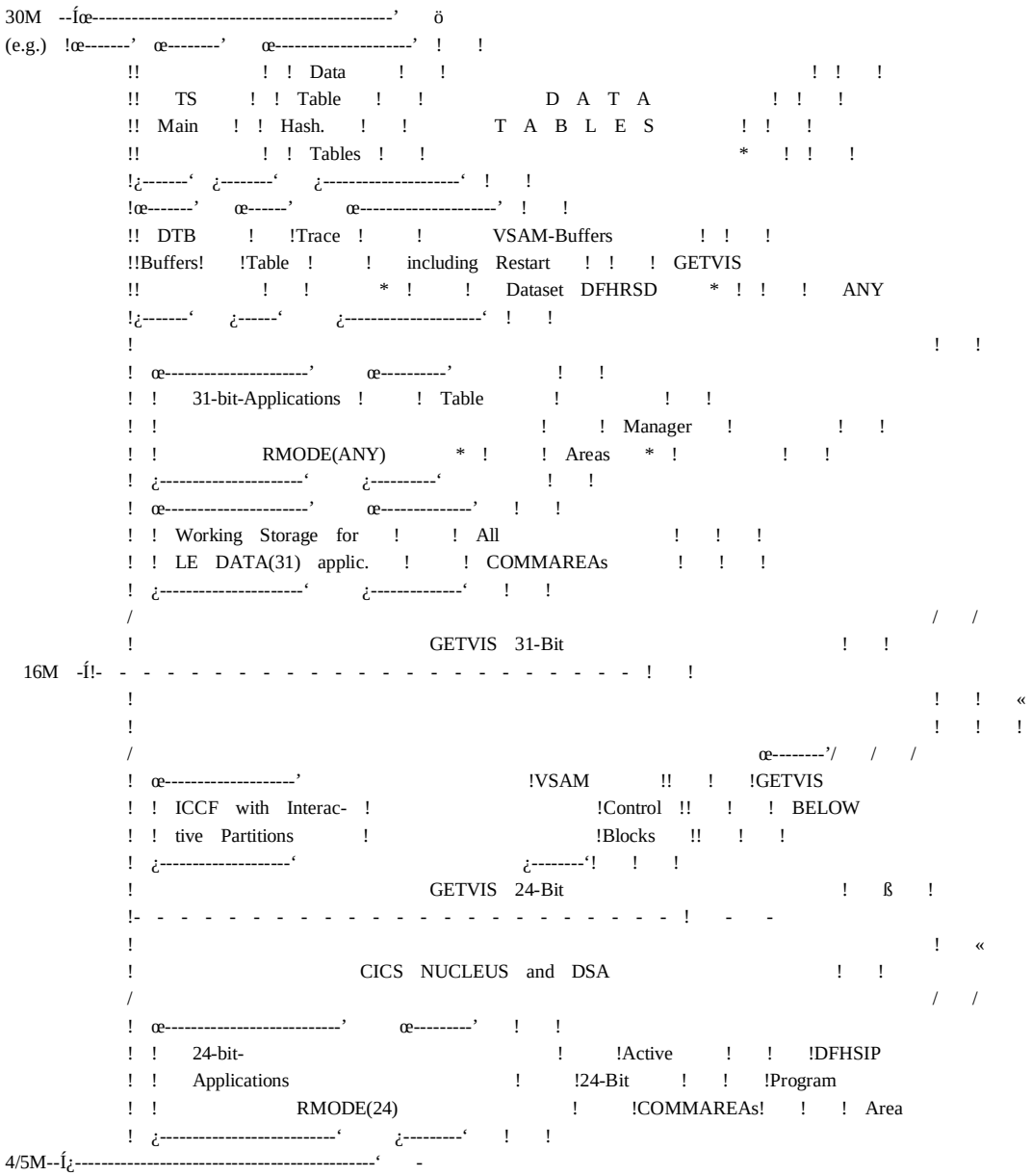
Virtual Storage Evolution

Simplified CICS Partition Storage Layouts



- More and more CICS areas moved above the line (VSCR)
- CICS storage areas grow (DIM), your capacity grows
- More storage -- Less I/O -- Higher txn volume

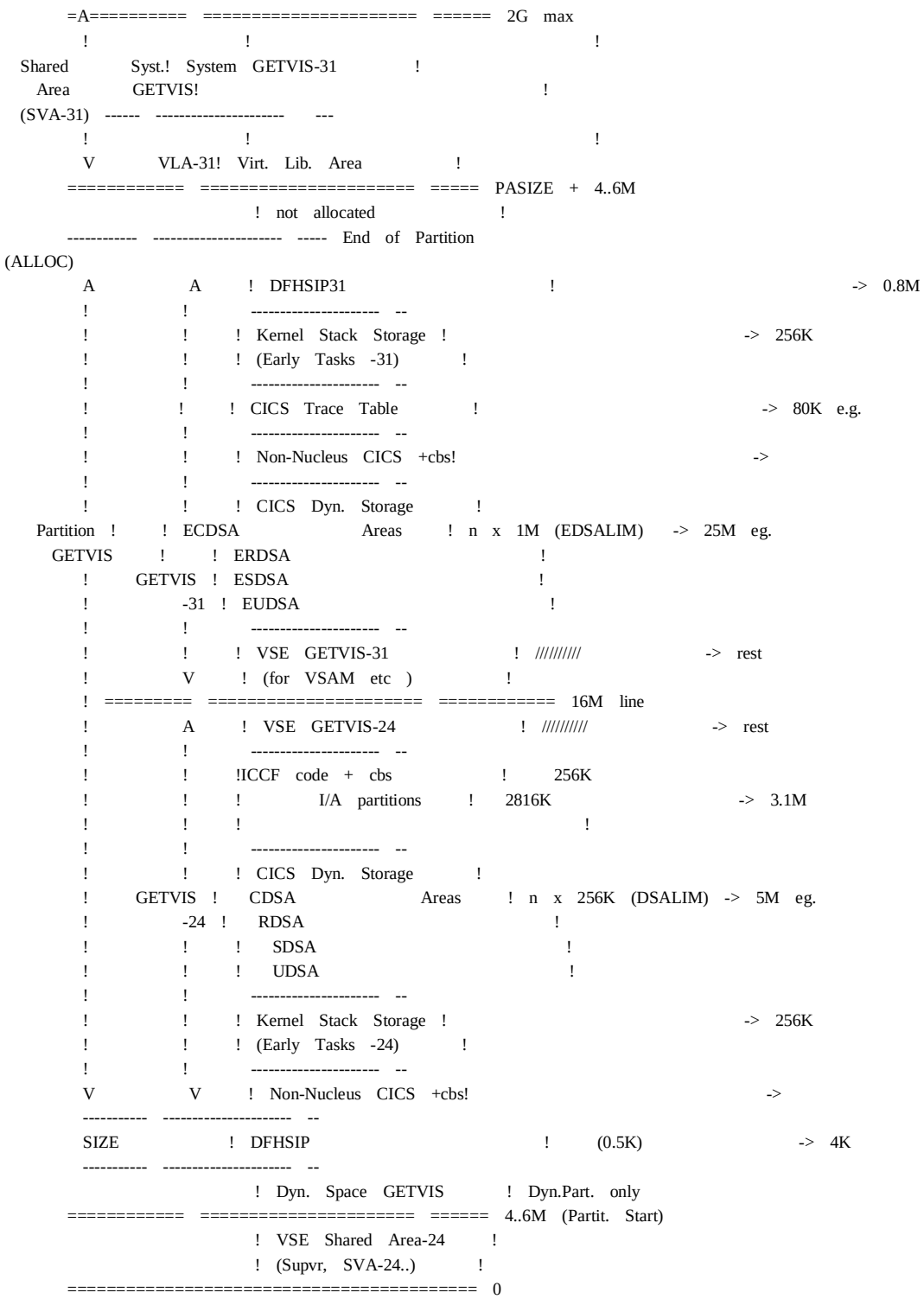
CICS/VSE 2.3 Storage Layout



* GETVIS space requested from long-lived subpool 0,
not from short-lived GETVIS subpool ICISS (Refer to GG24-4185)

\- CICS/VSE Message Table is in VIO storage, accessed via VPOOL

CICS TS Storage Layout



Dynamic DSA Management

- **Only total (max.) sizes are specified**

  SIT DSALIM for 4 DSAs below
  SIT EDSALIM for 4 EDSAs above the line

  These overall limits can be varied online via
  CEMT SET DSAs ! Dsalimit(nnM) ! EDsalimit(nnM)
  CEMT SET SYSTEM
  CEMT INQUIRE SYSTEM (by overtyping)

- **Individual DSA sizes are determined by CICS TS and are dynamically varied**

  (except SIT overrides for fixed sizes are used)

DSA type	DSA name	Description. Used for...	Key	
CICS	CDSA ECDSA	All non-reentrant CICS-key pgms, ctl. blocks, task-lifetime storage	CICS	
Shared	SDSA ESDSA	All non-reentrant user-key pgms, all storage obtained for pgms via EXEC CICS GETMAIN w/ SHARED option *1	CICS	
User	UDSA EUDSA	All user-key task-lifetime storage (no pgms)	CICS *1	
Read only	RDSA ERDSA	All reentrant pgms and tables (RENT attribute) (Key=0 is used)	CICS *2	

- All DSAs above are in multiples of 1M,
  all DSAs below are in multiples of 256K ('extents')
- CICS-key is used, except for the conditions shown
  *1 User-key if STGPROT=YES
  *2 Read-only key-0 if RENTPGM=PROTECT (default)

- **Read Only Protection (RENTPGM=PROTECT): No more usage of SVAs required for this purpose**

  (Stops overwriting by programs not running in key 0)

  CICS TS Loader will automatically use the RDSA and ERDSA if a program is eligible (i.e. link-edited with the SVA attribute)

  This is independent of Subsystem Storage Protect]

CICS TS Dynamic DSA Management ...

CICS DSA Organization

Each DSA consists of 'DSA subpools' and of a 'storage cushion'.

Each subpool is dynamic in size and consists of 4K pages

- PGSIZE specification in SIT was dropped
- No pages are shared between DSA subpools
- Note that it is no more required (in exceptional cases) to analyze the/a PAM (Page Allocation Map) in a Dump, since space requirements of individual DSAs (and subpools??) are reported in detail.
- Allocation increments in a subpool is 4K in general, only the EUDSA uses 64K

CICS TS uses about 180 DSA subpools

Mostly in DSAs above the line

DSAs are only increased or reduced by 256K/1M

The maximum sizes are GETVISed at CICS TS startup.

- **No more frequent VSE GETVISes**
  - **GETMAINs mostly satisfied w/o VSE GETVISes**
- **No more such GETVIS fragmentation problems (with long GETVIS searches)**

CICS TS Handling of DSA Shortage

„ Program Compressions now also above the line and done incrementally, before SOS is entered

CICS deletes RES=NO programs no longer in use in the last period of time (earlier but smarter than before, since it tries to keep more recently used programs).

Programs are only reloaded if required later.

í CICS/VSE never did pgm compressions above the line

This resulted for CICS/VSE in

- a different meaning of RES=NO for 31-bit programs (they stayed above the line after first usage)

- DFH0505 messages 'CICS IS SHORT ON EXTENDED STORAGE', when CICS partition size was say less than 30M

„ SOS (Short-On-Storage) Conditions

If Pgm Compression is not sucessful... CICS TS goes SOS

(If a task is suspended, or any DSA Largest Free Area is too small AND no free 'DSA extent' is available)

1. Tasks requesting new storage will not be resumed
2. Txns are abended if SPURGE(YES) specified (storage is freed)

No I/Os are required for these actions.

í CICS TS deadlock detection applies and thus may avoid a stall condition for a txn

„ CEMT INQUIRE SYStem

Gives current SOS status:
SOSStatus (NOTSOS!SOS!SOSBELOW!SOSABOVE)

Also, CEMT Inquire DSAs may be used.

CICS TS Kernel Storage

Ù CICS Kernel storage

Contains control blocks and data areas required to manage system and user tasks.

Static and Dynamic tasks

Task type	Storage	Location	# tasks	
Static	pre-allocated (e.g. user txns)	CDSA,ECDSA	MXT+1	
Dynamic	not pre-allocated (e.g. system tasks)	CDSA,ECDSA	8 and more *1	

*1 Number depends on CICS startup and functions used.
Not controlled by MXT meachanism.
Storage for a dynamic task is freed after use.

- Most of task storage is in the DSAs, but small amount is obtained from VSE at startup time ('early tasks').

Storage per task (static and dynamic)

Storage	Location
2K stack	24-bit, CDSA
12K stack	31-bit, ECDSA
896 byte kernel task entry	31-bit
4K extension stacks *1	24-bit, CDSA
4K extension stacks *1	31-bit, ECDSA

*1 1 per 10 tasks in MXT, pre-allocated

CICS Kernel Storage requirement

KE storage below: (MXT+4) x 2K + ES x 4K
above: (MXT+4) x 12K + ES x 4K
(ES=INT((MXT+4)/10))

e.g. MXT=60: --> ES=6 --> 152K below
792K above

í Do NOT overspecify MXT. May cause VS constraints since pre-allocated

Refer to the SIT MXT charts in the SIT part of this document, showing MXT startup variations.

CICS TS Message Table(s)

„ **CICS Message Table(s) for CICS/VSE were in VIO**

„ **CICS Message Table(s) for CICS TS is above the line and reside in EDSA, or in SVA-31**

Modules	DFHMET1E	262K	
	.....1C	254K	
	.....1G	254K	
	.....1K	271K	in total 1M

Use the SVA-31 for them

- for multiple CICS TS partitions
- only if SVA=YES can be specified

CICS TS Areas Required Below the Line

Areas which must stay below the line

- **Few VSAM control blocks, SAM control blocks and buffers etc.**
- **24-bit applications**
- **ICCF code and Interactive Partitions**
- **Selected CICS areas**

  - EIB and TWA of the TCA (by default)

  (EXEC Interface Block and Transaction Work Area of the Task Control Area = 'task life-time storage')

  -> For transactions/applications running totally in AMODE-31 (includes user exits), change that value via DEFINE TRANSACTION TASKDATALOC(ANY)

  Programs that use services which do not support 31-bit addressing:

  - Some code for TranData, Journaling, Txn Dumps, AUXTrace

  All programs with DTF or ACB access to files:

  - Very minor CICS Tables

  Journal Control Table (JCT)
  Terminal Control Table (TCT, for non-VTAM)
  Terminal List Table (TLT)

CICS TS VS -Revisited-

Background Info

- CICS manages DSA virtual storage in 4K page units
- TOTAL virtual storage can be much larger than real storage

  CICS and VSE keep only active data in real memory

  Inactive data can be paged out inexpensively

  VSAM LSR buffers need about a 1 to 1 virtual to real ratio

  (All buffers of an LSR subpool must be searched before a READ miss can be detected)

  For a CICS Data Table, dep. on access pattern, a ratio >1 to 1 may be appropriate

  Access is direct, via key.

Changes vs CICS/VSE

- Much more CICS nucleus now in 31-bit area
- 24-bit and 31-bit dynamic storage areas (DSAs)
- Use of data spaces for Shared Data Tables

  Beneficial even if not actually shared between CICS TS partitions

CICS TS VS Recommendations

- **Additional storage capabilities should be exploited: Exploit 31-bit storage where possible**

- **Be generous in your virtual storage allocations: Overspecify storage to start**

- **Ensure enough GETVIS-24 and -31 is available for VSAM, etc...**

  Watch out for 'GETVIS creep' from application errors

- **Ensure sufficient storage is available in DSA and EDSA**

  - **Make DSALIM and EDSALIM big enough**

    DSALIM must be at least 2M.

    Be very generous with EDSALIM (10M minimum may be too small)

  - **Use RES=YES for 31-bit programs**

    to avoid re-loading

  - **Identify maximum requirements for each DSA area (high water marks)**

    (DSAs below the line have to be more carefully observed)

  - **Monitor DSA and EDSA areas in the CICS TS Statistics (DFH0STAT or DFHSTUP)**

    - 'Times went short on storage' = ZERO
    - 'Times request suspended' = near ZERO
    - 'Times cushion released' = ZERO or very small

CICS TS VS Recommendations ...

Recommendations for CICS TS partition size

At comparable load to CICS/VSE 2.3 ...

Select partition size at least about 5M larger (above the line) than before with CICS/VSE 2.3

Compensate the move upward of CICS code and control blocks.
Use the better storage management above the line for more contingency.

Add 1M, if CICS Message Table(s) not in SVA-31.

Consider that in CICS TS, Data Tables moved into data space

At least 11M above the line, plus VSAM buffer GETVIS-31

More Hints

- **Larger virtual storage allocations can be more efficiently managed than smaller allocations**
- **Paging should be low**

  Paging does not result from excessive virtual allocation, but from excessive real storage use

- **(More on) Monitoring of CICS TS Virtual Storage**

  - CEMT Inquire DSAs
  - CICS Storage Statistics in DFHSTUP
  - CICS Formatted Dumps
  - VSE GETVIS commands
  - VSE IUI displays

PART D.

CICS TS More New Facilities

- VSAM LSR Setup
- Shared Data Tables
- Subsystem Storage Protection
- Security

CICS TS VSAM LSR Buffer Pool Setup

Background Info

- **CICS uses VSAM LSR Buffering to**
  - **Save/Share storage for buffers and control blocks**
  - **Extend VSAM lookaside to ALL buffers containing data from a given file**
  - **Save physical I/O activity**

    Less I/Os vs NSR due to lookasides.

    No I/Os saved for SHROPT4 due to required buffer invalidation after random READ.
  - **Save CPU time that would be spent in I/O mgmnt**

    But subpools should not become too long (so far)

- **VSAM LSR file management is recommended and is default for CICS VSAM files**
  - **Up to 15 LSR pool-IDs can be defined**

    Each with subpools (set of buffers with same CI-size)
  - **Separate (sub) pools for data and index buffers possible (CICS TS, new)**

    Specific files can be set to LSRPOOL=NONE

CICS TS VSAM LSR Buffer Pool Setup ...

More Info

- **LSR should always be used**

- **Exceptions:**

Deadlocks are possible for txns with concurrent BROWSE and UPDATE,
but this can be avoided by redesign
(Refer to CICS Application Programming Guide).

- LSR allows concurrent BROWSEs
- LSR prohibits concurrent UPDATE of a CI
- LSR prohibits READ/BROWSE when a CI is held for update

The new deadlock detection detects such a deadlock earlier.

- **NOTE to VSAM Seq. READ-Ahead:**

(not a decision criteria LSR vs NSR)

- Not performed and not reasonable for LSR files
- Only applicable for sequential access (BROWSE)

- Heavy BROWSE activity (occurs frequently) dominates other activities

  - via stealing VSAM buffers (except when in own unique subpool)
  - via fast logical I/O w/o WAIT
  - often crosses more than 1 CI boundary

Changes vs CICS/VSE

- **CSD management for file definitions is new:**

  CEDA DEFINE FILE and SUBPOOL via RDO

- **CSD management for LSR pools is also new:**

  Extra (sub)pools for DATAxxx and INDEXxxx

CICS TS VSAM LSR Buffer Pool Setup ...

Recommendations

- **Use LSR, unless impossible for a file**

  Consider application redesign to avoid deadlocks, if critical

- **Use multiple pools, separate DATAxxx and INDEXxxx**

  Requires use of CEDA DEFINE FILE and LSRPOOL

- **Separate files with heavy BROWSE from other files and from each other**

- **For each pool, define DEFINE LSRPOOL entries for**

Maximum key length

Set KEYLENGTH to required amount:
Saves storage, startup time

Number of strings

Set STRINGS to 'max ever used' plus small extra.
Ensure enough exist: check statistics to avoid huge excess

Number of buffers of each size (= subpool size)

to avoid startup delays and to optimize buffer pools.

Separate the (sub)pools for DATAxxx and INDEXxxx

For good hit ratios:

- Lookasides > 10 times READs (higher is better)
- Consider moving files to a new pool when number of buffers exceeds 100 to 150

Cont'd

CICS TS VSAM LSR Buffer Pool Setup ...

Recommendations (cont'd)

- **Start with automatic calculation of LSRPOOLs if desired**

  Specify LSRPOOL details for optimal setup and faster startup

- **Exploit more pools, with smaller subpools (fewer buffers) each, rather than fewer pools with more buffers each**

  Try not to oversize subpool size

- **Monitor status with CICS LSR Statistics**

  From DFHSTUP or DFH0STAT statistics

  - Statistics format changed
  - Data expanded and clarified

- **For performance trials .... LSRPOOL definitions can be changed dynamically (when CICS TS is up, using RDO)**

  - if all participating files are closed
  - if enough GETVIS was provided

CICS TS Shared Data Tables (SDT)

Background Info

- **Benefits of Data Tables in general:**

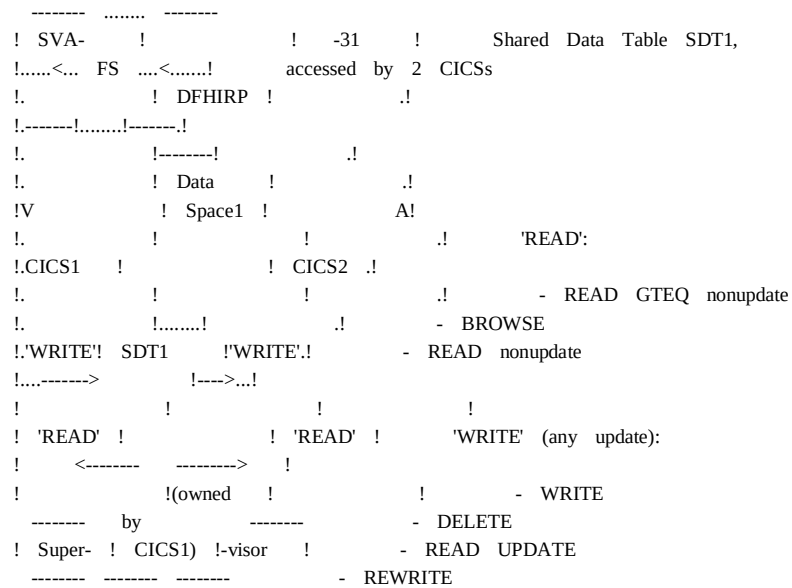
  Much shorter pathlength than in VSAM LSR

  Refer e.g. to CICS/VSE 2.2 Data Tables charts in
  'IBM VSE/ESA 1.3/1.4 Performance Considerations'

- **SDTs replace the 'basic' Data Tables in CICS/VSE 2.2/2.3, with significant enhancements:**

  Can also be used for imprecise key (GENERIC or GTEQ) and BROWSE
  Can be shared between CICS TS partitions
  Functional benefits (smaller dumps, security..)
  Function Ship (FS) overhead only for 'remote updates'

- **Example**



CICS TS Shared Data Tables (SDT) ...

Performance View

- **Traditional FS overhead only for 'WRITEs' from CICS2**

  No more FS overhead for READs from CICS2 due to the use of cross-memory services

  READ/WRITE performance for CICS1 is similar to the former CICS Data Tables in CICS/VSE

- **A SDT resides in a data space (defined SINGLE), owned by the FOR, extended in 2M increments**

  A separate data space per FOR, used for all owned SDTs

- **Data Table key ranges may have 'gaps', as before**

  For requests within the gaps, the request is handled as a 'normal' VSAM request to access the KSDS source data set via VSAM LSR

- **SDT access control blocks reside above the line**

SDT Access control blocks			
Table-entry storage	Keylength +9	byte 1/record (mult. of 8 byte)	
Index storage	5.. 50..	76 byte 1/record	

- Storage is allocated in multiples of 32K
- SDT access storage, naturally, only required in the SDT owning CICS

As an example, consider a KSDS file with KEYLENGTH=24 byte.

The total required size for access info in 31-bit is:

24+9 => 40 + 50 about 90 byte per record

- **Actual SDT usage**

  Can be determined from the statistics record DFHA17DS, and is part of the DFHSTUP output.

CICS TS Shared Data Tables (SDT) ...

Changes vs CICS/VSE

- **More flexible, so often more beneficial than before**

  Now also benefits for Browse and generic key READs

- **FS overhead for 'remote' READs are avoided**

- **Still 2 types exist:**

  CICS-maintained data tables (CMT)
  User-maintained data tables (UMT)

  Usually, CMTs are used, which automatically care for propagation of updates.

  Refer e.g. to CICS/VSE 2.2 Data Tables charts in 'IBM VSE/ESA 1.3/1.4 Performance Considerations'

Measurement Results

Refer to the VSAMTEST measurements, in the measurement part: 'SDT vs VSAM LSR'.

CICS TS Shared Data Tables (SDT) ...

Recommendations

- **Use SDTs for frequently READ or BROWSEd, not too big VSAM KSDSs**

- **Consider using CICS-maintained Data Tables**

  - no change in application
  - integrity across CICS restarts
  - if (only) journaling of updates is required

- **Make sure that SDT support is enabled for all CICS TS partitions that should/could benefit**

  A) 'Nonshared SDT'
  - DEFINE FILE as a Data Table
  - DFHDTSVC, DFHDTSAN, DFHCSEOT in SVA

  B) 'Shared SDT' (in addition)
  - ISC=YES (+ MRO links) for remote and SDT owning CICS
  - DFHIRP for MRO in SVA-24

- **Make sure Data Spaces can be created**

  Enough BG ASI SYSDEF DSPACE,DSIZE=nM set aside from VSIZE

- **Make sure Data Spaces can be extended**

  Enough DSIZE set aside out of VSIZE.

  DSPACE from CICS TS startup is only used/required for CICS TS as a VTAM application, not used for the SDT data space

- **Use big Data-CI sizes**

  Reduces Data Table load time

- **Monitor the Data Table usage and benefits**

  Use the CICS Data Table Statistics, discussed in the CICS TS Performance Guide.

CICS TS Shared Data Tables (SDT) ...

CICS SDT Recommendations (cont'd)

- **Check whether the number of LSR buffers can be reduced, when files are defined as SDTs (together with LSR)**

  When all (or most) of READ data accesses are satisfied by the SDT, there is no need for many LSR buffers

- **Assure that you do NOT 'overdim'.**

  Any increase of paging may be more harmful than the benefits an SDT can bring

Refer also to

'CICS TS Shared Data Tables Guide', SC33-1668-00

CICS TS Subsystem Storage Protection

CICS Subsystem Storage Protection (SSP)

Background Info

- **New integrity function for VSE/ESA users**

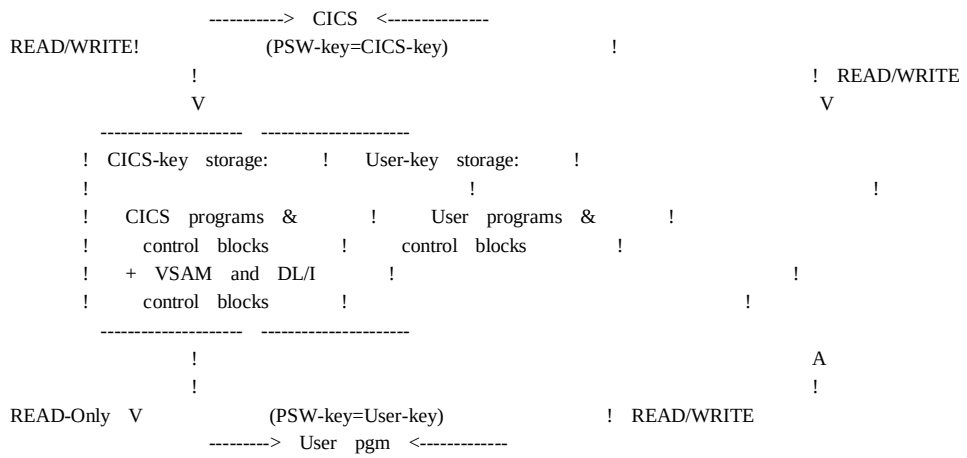
  Protects CICS nucleus code from accidental overwrite by user pgms.
  Also VSAM RPLs, ACBs, EXlists are protected for files explicitly OPENed by CICS, PLUS DL/I 1.11 system resources

- **Can help identify storage violations**

- **Program causing violation is abended**

  Identity of violator and cause can be found more readily than before

- **More thorough test and validation**



- **Usage of different storage keys**

CICS-key = partition-key
User-key = 'public-storage-key' = 09

No subsystem storage protection for CICS-key items in F4 (F4 has partition-key 09)

CICS TS Subsystem Storage Protection ...

CICS TS SSP ...

- uses the ESA/390 Storage Protection Override facility
- is activated via SIT STGPROT=YES
  and defined per program (new)

  DEFINE PROGRAM EXECKEY=CICS or USER

  This defines in which key CICS gives control to the program. Normally, USER should be selected (default), to exploit SSP.

  PLT, PLTPI, and PLTSD programs must run in CICS-key

- does NOT prevent one application overwriting another

  This function would be 'Program Isolation' with 1M user storage increments (OS/390)

Performance View

- Separate DSA areas used for CICS and USER key,
  using different hardware storage keys
- Increased integrity and availability

Changes vs CICS/VSE

- SSP is a new facility in CICS TS

CICS TS Subsystem Storage Protection ...

Measurement Results

- **Performance cost of SSP**

  Generally very small extra CPU time

  Correct PSW and storage keys must be set when control is transferred.

  Performance runs with DSW showed

  NO measurable CPU-time impact (+-0%)

- **Many performance benefits are indirect**

  Due to higher availability and integrity

Recommendations

- **SSP should be exploited in test environments**
- **Use it in production to identify violations**

  except any vendor program would not tolerate it

  No cost even for really stable production systems
- **See STGPROT SIT parameter description**

CICS TS Security

Background Info

CICS TS issues RACROUTE calls for Security

„ Basic Security Mgr (BSM)

Provides Signon and Transaction Security

- Included in VSE/ESA 2.4 Base
- An enhancement of VSE/ESA Access Control
- Addt'l Access Authority Checking if SYS SEC=YES
- Appears to CICS like an 'ESM'

„ External Security Mgr (ESM)

Also provides Resource and Command Security

- Activated via SYS ESM=phasename
- Vendor programs (CA Top Secret for VSE/ESA 1.3.0, BIM-ALERT/VSE, BIM-ALERT/CICS or equiv.)

Performance View

IPL Parameters:

SYS SEC=RECOVER no security manager, for critical restart only
SYS SEC=YES CICS and Batch (ACLR like) security
SYS SEC=NO only CICS security - no Batch security

CICS Parameters:

SIT SEC=YES full external security
SIT SEC=NO no external security manager

Parameter Combinations:

SYS SEC	SIT SEC	CICS Security Functions	Comment	
NO	NO	None, no BSM. no ESM	Unusual	
NO	YES	BSM	Mostly used	
YES	YES	ESM or BSM (depends on SYS ESM=)	Secured Envir.	
YES	NO	No BSM, no ESM	Unusual	
RECOVER	YES	Just for recovery	Only use once	

- SECTAB Security is for Batch only
- SECTXN Security is for CICS TS (BSM) only

CICS TS Security ...

Security Tables:

	Used for	Location	Size	Comment	
DTSECTAB SYS	SEC=YES (Batch)	System GETVIS-24	2.5K		
DTSECTXN	BSM (CICS TS)	System GETVIS-31	8K	Contains shipped txns	300

Changes vs CICS/VSE

Internal Security in CICS no more supported

Measurement Results

SYS SIT SEC	CPU util	NPS	txn/sec	Av RT (sec)	CPU-time /txn (ms)	ITR-ratio	
NO NO	36%	0.290	20.33	0.073	17.92	1.00	
NO YES	38%	0.290	20.37	0.073	19.10	0.94	
YES YES							

- Standard system setup and environment
- IO/txn here very constant at 4.3.
  Here no increase of IO/txn, since no LOGONs occured during measurement interval.

+6% more CPU-time per txn

by SIT SEC=YES (SYS SEC=NO), i.e. BSM security

Note: All DSW measurements are done in stable status, w/o new LOGONs.

Nevertheless this example is shown here.

More Info

For more info refer to

'CICS TS Security Guide' SC33-1942

PART E.

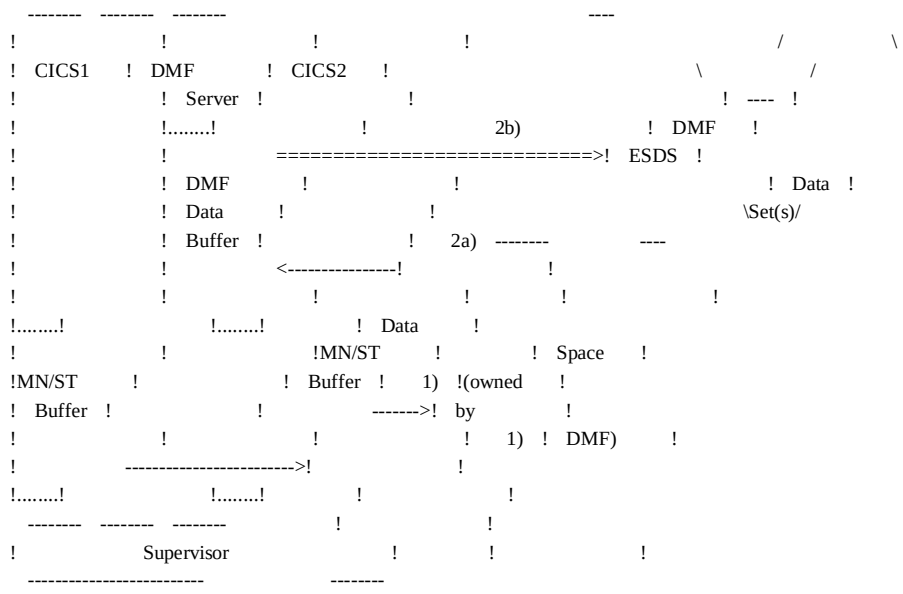
CICS TS New Operational Aspects

- Data Management Facility
- Statistics
- DFH0STAT
- Monitoring
- Performance Monitoring
- SVA Content
- Startup and Shutdown
- DL/I VSE 1.11

CICS TS Data Mgmnt Facility (DMF)

Background Info

DMF was written to provide a similar facility to SMF provided by MVS



„ Data Collection

1) The Statistics and Monitoring domains (ST/MN) collect data (in SMF format)
and let them being moved into the DMF owned data space (which exists after DMF was started).
Cross Memory Sevices are used by CICS TS to call DMF routines.

Optionally, also user output journals can be written to DMF, via JCT entries, but w/o journal read capability.

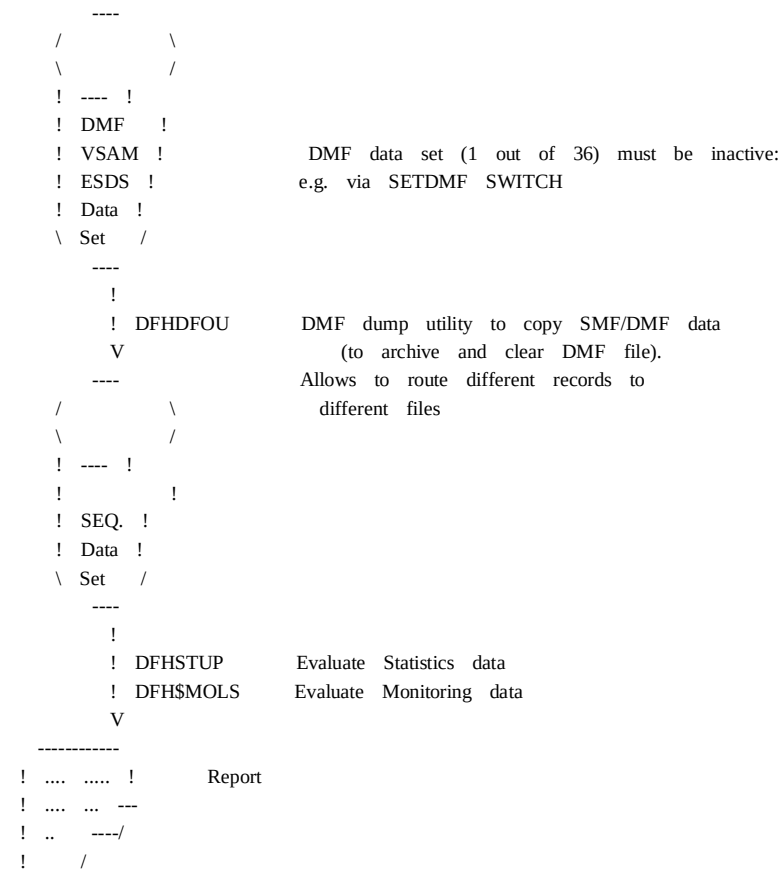
2) The DMF Data Handler ('DMF'), running in another static or dynamic partition, writes these data to the currently active DMF data set.

„ Data Evaluation

Refer to next foil

DMF Data Evaluation

DFHSTUP for Statistics data
DFH$MOLS for Monitoring data



DMF Background Info (cont'd)

Some DMF Commands

?	Lists general operating instructions
SETDMF FLUSH	Writes current DMF data space content to the VSAM buffers of the currently open DMF data set
SETDMF SWITCH	CLOSEs (no FLUSH included) VSAM file and OPENs next VSAM ESDS file in order
SETDMF SHUTDOWN	FLUSHes, CLOSEs, Shutdowns DMF, and releases the data space
SETDMF NOACTIVE	Stops DMF to accept data to be put into the data space, but leaves it up
SETDMF ACTIVE	Starts DMF to accept data to be put into into the data space (if NOACTIVE)
SETDMF DISPLAY O	Displays current settings for DMF
- Refer to 'CICS TS Operations and Utilities Guide'	

The DFHDMF initialization table (DFHDMFM macro) contains e.g.:

INTERVAL=mmss	Interval for writing data to DASD (0001-5959, default is 3000 = 30 min 00 sec)
SIZE=nnnn	Size of the shared DMF data space in MB. nnnn ranges from 4 to 2048 (MB).
USAGE=xx	Threshold to start writing of data (data space xx% full, default is 50)
SUFFIX=xx	Suffix of the DMF table to be used
STATUS=	ACTIVE starts recording when DMF is started (default). INACTIVE needs SETDMF ACTIVE cmd.
...	

DMF Background Info (cont'd)

„ Required Settings for Statistics from DMF

Required settings to get CICS TS statistics data into DMF:

Type of Statistics	CICS TS Setting
CICS Interval	STATRCD=ON
CICS Requested	CEMT PERFORM STATISTICS
CICS Shutdown	none

CICS TS and DMF Interaction

„ In order to accept data...

DMF must be already started before CICS TS AND must be ACTIVE

If CICS TS is started before DMF ...

- it will display msgs DFHST0103 and DFHMN0101 once and continues processing

- CICS TS permanently tries under the cover to put the records into the DMF data space

To start collecting records when DMF is started later ... the following is required within CICS TS:

- Statistics: No activity req'd

- Monitoring: A data description record has to be provided by running the DFHMNDUP utility in Batch

CICS TS Data Mgmnt Facility (DMF) ...

Performance View

- **A separate partition is required for CICS DMF**

  Slight impact on VSIZE, very minor impact on real storage (DMF can only be started once per VSE system).

- **A separate partition means potential CPU-overlap (n-way)**

  But DMF CPU-time is very low: 'just sitting around'

- **Data are written by DMF to DASD**

  - after a timed interval
  - when DMF data space has become x% full
  - when DMF is shut down or SETDMF FLUSH is issued

  - **No records are immediately forced to be written to DASD**

    Thus, NOT suitable for integrity journals.

- **All physical WRITEs to DMF files are done asynchronously**

  and do not delay production activity in CICS TS

- DMF files are VSAM ESDS, and REUSABLE with a maximum record lenght of 32K-1. Spanned records are allowed

CICS TS Data Mgmnt Facility (DMF) ...

Changes vs CICS/VSE

A totally new concept, but similar to the SMF record concept in CICS/ESA in MVS.

Measurement Results

Overhead of collecting CICS statistics

Early measurements with a CICS Online workload ('DSW', see separate part) showed about 1.5% more CPU-time required per txn.

Recommendations

- **In any case, get the DFH0STAT statistics printout, NOT requiring DMF**

- **Parameters for DMF partition startup**

  - **Partition size**

    Uncritical, select e.g. 2-5 MB.
    Suggest to use a dynamic partition
    (since any CICS is a long running jobstep)

  - **EXEC statement**

```
// EXEC DFHDFSIP,SIZE=DFHDFSIP,OS390
SUFFIX=xx
```

- OS390 as the last parameter is always required, since OS/390 functions are used

- NO DSPACE specification required/used to limit the maximum size of the DMF data space (DSPACE is only used by VTAM and VTAM applications)

- SUFFIX specifies the DMF initialization table

CICS TS Statistics

Background Info

As already addressed...

- **All Statistics (and Monitoring) data now are ...**

  - put into a data space, owned by the CICS Data Management Facility (DMF)
  - written by DMF to DMF data sets

  DMF (discussed earlier) runs in a separate partition.

  Reporting is done via DFHSTUP (DFH$MOLS for Monitoring)

- **In addition, statistics can be obtained directly from DFH0STAT**

  A COBOL program, NOT requiring DMF

Changes vs CICS/VSE

- **CSTT txn was replaced by new CEMT commands**

```
CEMT  Inq    STat          Get status of statistics collection
      Set    STat          Enable users to set collection options
      Perf   STat..        Write all/selected collected data immediately
                                 to the DMF data space
                           ..    Reset all statistics (RESET)
```

  NOTE:

  If no DMF is up and active, DMF data are lost

- **New STAT txn available, to directly get statistics (DFH0STAT)**

CICS TS Statistics ...

Ù Ways of producing statistic reports

„ CICS DFHSTUP

Requires DFSORT/VSE or equivalent sorting program

„ Sample statistics program: DFH0STAT

To get info on current statistics with the new DFH0STAT program

„ Your own program

Ù Ways of collecting statistic records

Writing statistic records to a DMF data set

„ Interval

Statistics must be on via SIT STATRCD or CEMT SET STatistics

„ End-of-day

Always done at CICS shutdown

„ Requested (or DFH0STAT via STAT txn)

„ Requested reset

„ Unsolicited

Ù Interpreting CICS Statistics

Refer to 'CICS TS for VSE/ESA Performance Guide', SC33-1667, Appendix A: 'CICS Statistics Tables'

CICS TS DFH0STAT Sample Statistics

Ù Sample Statistics program: DFH0STAT

A sample program in COBOL, not requiring DMF, called by the STAT txn

- to get current values of CICS statistics data

- Spooled to SYSLST, or
- Directed to CICS TempStor, for browsing with CEBR

- In FULL or ABBREVIATED form
  (We suggest to use the abbreviated form)

- to serve as a sample for any user written statistics program

Ù CICS TS setup can be such that DFH0STAT is automatically obtained as SYSLST job, when CICS is shut down (w/o IMMED)

Ù BUT:

- DFH0STAT includes all resettings

See next foil

- DFH0STAT does NOT show a full picture

Autoinstalled resources are NOT included, if deleted meanwhile. Same as in CICS/VSE Statistics.

í DFH0STAT does NOT make DFHSTUP obsolete

CICS TS DFH0STAT Sample Statistics ...

Ù DFH0STAT Report

- is very similar data as compared to CICS/VSE statistics and sufficient for nearly all cases

- is very similar but NOT as detailed as DFHSTUP, but wording and formatting is slightly different.

DFH0STAT is shipped as COBOL/VSE source program.

DFH0STAT data reflect the resetting of statistics via
- CEMT Perform STatistics RESetnow
- crossing midnight
- expiration of an interval for automatic stats

Ù DFH0STAT Data

- System Status, Monitoring and Statistics param.

Settings
Status
...

- Transaction Manager and Dispatcher

Total txns
Times at MXT
...
Peak Tasks
Current Tasks
TCB CPU Time
...

- Storage statistics for all 8 DSAs

Current Size
Current Used
Peak Used
...
Times Short-On-Storage
GETVIS sizes (BELOW!ABOVE)

- Loader

Total Loads
Avg Load time
Pgms removed by compression (per DSA)
Program Storage

CICS TS DFH0STAT Sample Statistics ...

Ù DFH0STAT Data (cont'd)

- Transactions

Tran-ID Tran-Class Pgm-Name Static/Dynanmic Location/Key
Counts ...

- Transaction totals

Tran-Count Attach-Count
... for any of the 4 Location/Key combinations

- Programs

Pgm-Name Dat-Loc EXEC-Key Times_used Times_Fetched... Pgm-Loc

- Programs totals

#pgms per language
#pgms per DSA
...

- Temporary Storage

Extensive TempStor stats

- Transient Data

Extensive TranData stats

- LSR pools

Usual LSR stats

- Files

Logical request counters
Physical I/O counters

- Data Tables

...

CICS TS Monitoring

Background Info

- **Monitoring data are written by DMF to data sets in SMF 110 record format**

  Record subtype 1 is for monitoring, 2 is for statistics

Performance View

- **All monitoring code and control blocks located above**
- **All user storage above the 16M line**

Changes vs CICS/VSE

- **6 Monitoring SIT entries (MN...=) available,**

  replacing the MON= options

- **Previously, CICS Monitoring used CICS Journaling**

  Now DMF is used.

Measurement Results

- Maybe, performance monitors for CICS TS could intercept before monitoring data are written to data space or data sets.

  The CICS TS Customization Guide shows what Monitoring exits are provided. In this case this would be XMNOUT or XSTOUT.

- CICS Monitoring overhead by writing monitoring data to journal data sets was roughly up to 15% for CICS/VSE

- Measured CPU-time increase was 11% for DSW (see SIT MN discussion)

CICS TS Performance Monitoring

Performance Monitoring and Trouble Shooting are related activities.

Any sequence of analyzing and of narrowing down a problem area is reasonable and brings you faster and closer to a solution.

For more hints refer also to the document
'IBM VSE/ESA Hints for Performance Activities'

Ù VSE System Info

Before you invest too much time in a detailed analysis ...

Check whether the problem may be solved more efficiently by changing VSE environment or parameters or setup.

Any VSE system performance tool may be used, including

- VSE Job Accounting

(be it via the $$JOBACCT output on SYSLST
or via the IUI System Activity Display)

- the QUERY TD function

- SIR SMF (for rough msec/IO)

- VSAM LISTCAT

(before OPEN and after CLOSE of a file)

- VSE System Performance Monitors

e.g. EXPLORE/VSE,	from Computer Associates	
TMON VSE	from Landmark	
Minimon	from The Minimon People	
.....	from BIM	??
AUTOMON	from	??

CICS TS Performance Monitoring ...

Ù CICS Info

For CICS specific activities, the following tools are recommended in the sequence shown

- **CICS Statistics**
- **CICS Performance Monitors**
- **CICS Internal and Auxiliary Trace**

In any case ...

a CICS performance monitor is helpful and much more than nice-to-have for bigger CICS production systems

e.g. CA-EXPLORE for CICS	from Computer Associates
TMON/CICS	from Landmark
OMEGAMON	from Candle Corporation
Minimon	from The Minimon People

Minimon V1.2 supports also CICS TS (2000-03-21), refer to
http://www.minimon.com

í Refer to the vendors to get latest information

í Be aware that monitoring overhead exists and depends on specified monitoring options

CICS TS SVA Phases

Types of Phases to be Distinguished

Phases used in a CICS partition

1. CICS TS phases required in SVA

NOT affected by any SIT SVA definition

2. CICS TS phases (DFH*) optionally in SVA

Affected by SIT SVA definition

3. Other phases (e.g. LE or user pgms) optionally in SVA

Also affected by SIT SVA definition, as in CICS/MVS.

CICS TS Mandatory SVA Modules

1. CICS (VSE and TS) Modules Required in SVA

The following CICS modules are required in the SVA for FUNCTIONAL reasons

(provided function is used)

Name	Description	loc.	SVA CICS/VSE	SVA CICS	Appr. size(byte) TS
DFHIRP	Interregion communication	-24	12280	16304	
DFHIRW10	-" - work exit	-24		-	1260
DFHCSEOT	CICS EOJ cleanup (MRO, SDT)	-24	400	216	
DFHSCTE	Subsystem control table	-24	8	16	
Maximum functional requirement in SVA-24			12688	17796	
DFHCSVC	CICS SVC module	-31		-	2224
DFHDSPEX	DS domain- VSE POST stub exit	-31	-		200
DFHCDDAN	CPC dead data anchor block	-31	8		8
DFHDTSVC	Data Table support	-31	-	10424	
DFHDTSAN	Data Table anchor block	-31	-		32
Maximum functional requirement in SVA-31			8	12888	

These modules must (and are) used from SVA, independent of any other settings.

Ù **Critical for VSE systems with CICS TS AND CICS/VSE installed:**

í **Make sure only the CICS TS versions of these mandatory SVA phases are loaded into the SVA.**

Reason: The CICS/VSE versions of DFHIRP, DFHCSEOT, DFHCSTE, DFHCDDAN
do NOT work for CICS TS,
BUT the CICS TS versions also work for CICS/VSE 2.3.

2. CICS TS Modules Optionally Loaded in SVA

Many CICS modules may be loaded into the SVA (-24, -31) for PERFORMANCE reasons:

('Optional CICS modules')

- Saving real storage for multiple CICSs
- Filling up a required 1M segment below the line (SVA-24)
- Getting a faster CICS startup
- Protecting these modules from 'rogue' applications
  (before or w/o Subsystem Storage Protection, a side effect)

Refer to 'CICS TS System Definition Guide', SC33-1651-00, Appendix B

About 30 CICS TS phases with a rough total size of about 350 KB are eligible for SVA-24 only

Do e.g. a LIBR LD DFH*.PHASE of PRD1.BASE, and search for phases with SVA ELIG YES and R-MODE 24 (exclude the SVA mandatory ones).

-> This is NOT enough to compensate for a 1M segment rounding.
Look for additional other used (-24 only) modules (e.g. LE).
Do not fill that space with any 31-bit modules.

These modules are used by CICS TS only if SIT SVA=YES

PRVMOD exceptions are discussed later.

- **Notes of CAUTION:**

- **Do NOT load opt'l CICS TS modules into the SVA**
  (with a common name to CICS/VSE 2.3),
  if you also have CICS/VSE installed

- **SIT SVA=YES only can be set, if NO optional CICS/VSE modules are loaded into the SVA**

  Reason: Optional SVA eligible modules exist for both CICS/VSE and CICS TS with same name but DIFFERENT content.

Cont'd and refined on next foils

CICS TS Optional SVA Modules ...

More SVA Specifications for CICS TS

Instead of SVA=NO an alternate specification exists:

- **SVA=YES plus PRVMOD and DFH$SVEX.J**

  PRVMOD (a list of PRiVate MODules, i.e. not SVA shared)
  allows to avoid functional problems for CICS TS
  in case SVA=YES is set in a VSE environment with CICS/VSE:
  i.e. a list of modules for which 'SVA=NO is in effect'.

  Note, PRVMOD cannot be coded in a CICS TS SIT:
  it is used in PARM, SYSIPT, or as CONSOLE overwrite.

  DFH$SVEX.J in PRD1.BASE contains the PRVMOD parameter and the
  names of all SVA optional CICS TS DFH* modules with common name
  to CICS/VSE 2.3:

  Use e.g. * SLI MEM=DFH$SVEX.J,S=PRD1.BASE

  Allows CICS TS to use e.g. LE phases from the SVA, w/o using any ('wrong') CICS DFH* phase.

- **To start with, SVA=NO for CICS TS is the safest way for environments with both CICS TS and CICS/VSE installed**

- **You can use USESVACOPY(NO/YES) via RDO for a non-nucleus CICS module or a user pgm**

  (all phases that require a program definition via CEDA).

  The default is NO for all programs in the CSD,
  and means NOT to use the SVA copy, even if you specify SVA=YES.

- **PRVMOD and USESVACOPY 'exceptions' must be carefully used**

CICS TS Optional SVA Modules ...

Overview on optional Phases in SVA, used by CICS

Installed Environment	CICS TS and CICS/VSE	CICS TS only
Load optional CICS/VSE modules into SVA	Only allowed with SVA=NO	- in CICS TS (or SVA=YES with PRVMOD & DFH$SVEX)
Load optional CICS TS modules into SVA	Not allowed (CICS/VSE has no SVA=NO spec.)	*1 yes
Load non-CICS modules (e.g. LE) into SVA	yes	yes
Use of any SVA phase by CICS/VSE	yes	-
Use of SVA phases by CICS TS		
- DFH* phases	Not possible	*1 Possible (SVA=YES)
- non DFH* phases	Possible	*2 Possible (SVA=YES)

*1 No optional CICS TS modules are allowed in SVA, except those with unique names compared to CICS/VSE
*2 SVA=YES and DFH$SVEX.J is required in CICS TS

SVA use for CICS Coexistence -REVISITED-

When you have both CICS TS and CICS/VSE installed

Again, be cautious

- **You CANNOT load SVA optional CICS TS modules into the SVA, having same name as in CICS/VSE**

  (they would be used by CICS/VSE which would fail)

- **SVA=NO (shipped value) is required**

  SVA=YES is only allowed in a restricted manner (see PRVMOD)

- **You CAN load optional CICS/VSE modules into the SVA**

  BUT

  - they can only be used by CICS/VSE

  AND

  - you must assure that CICS TS will NOT use them (since it would fail).

    This condition is fulfilled if either in the CICS TS SIT

    - SVA=NO is specified

    or

    - SVA=YES is specified AND DFH$SVEX.J is used:

      * SLI MEM=DFH$SVEX.J,S=PRD1.BASE

- **SVA=YES plus DFH$SVEX.J is the only way to use e.g. LE-phases from the SVA (in CICS TS, when also CICS/VSE is installed)**

More Info on SVA Loading

When you only have CICS TS installed

Especially when >1 CICS TS partitions are used ...

- **Load all SVA-31 eligible CICS TS modules into the SVA-31**

  Make sure that SVA-31 is big enough (SVA PASIZE-31),
  in order to avoid that modules are 'downloaded' into SVA-24

More Information on SVA Loading

In 'VSE/ESA V2 Performance Considerations', Appendix A:

- **'Virtual Space Optimization in VSE/ESA'**
- **'General SVA Load Strategies for Performance Reasons'**

In 'CICS TS System Definition Guide', SC33-1651-00', Appendix B:

- **'SVA eligible CICS phases'**

CICS TS Principal Startup

Background Info

- CICS TS was ported from CICS/ESA for MVS 4.1
- MVS macros have been kept
- VSE/ESA 2.4 accepts only the TS subset of these macros:

  'Family API'
- Program is switched into this mode by

  EXEC xxxxxx,.....,OS390

  - 'OS/390 Emulation Mode'
- This partition must be the only one in an address space
- Select a partition size at least about 5M/6M larger than before

  Use much more and let CICS TS use whatever it needs of it (only costs VSIZE and thus page data set storage).
- Make sure enough data space is available

  Besides for VTAM, required if SDTs are owned
- CICS TS Startup

  EXEC DFHSIP,SIZE=DFHSIP,PARM='SI,SIT=...',DSPACE=3M,OS390

  - SIZE=DFHSIP results in EXEC-SIZE=4K only]
  - Make sure enough DSPACE=nM is used (3M, for VTAM use, only)

CICS TS Partition Startup and Shutdown

Background Info

- CICS Startup Time can be critical to availability
  - New features and functions available
  - Tuning can reduce startup time
- Shutdown time may not be as critical
  - Tuning can reduce shutdown time

Performance View

- Program Loading
- Dataset Opening
  - GCD, LCD options

Changes vs CICS/VSE

- GCD, LCD are additional CICS system datasets
  - Contain re-start information
- RSD now used for emergency restart only
- Review tuning information in CICS Performance Guide
  - General VSAM tuning applies
  - Use REPRO/DELETE/DEFINE/REPRO to reorganize or change

Recommendations

VSAM File Hints:

í Define GCD, LCD, CSD and RSD for performance

- Set DATA CI to 8K or so
- Set INDEX CI to a small value
  (1K for LSR, and NSR)
- Examine LISTCAT output after execution

- CSD is now defined in the SIT (no more in FCT)

í Set BUFND and BUFNI on DLBL statements for GCD, LCD and RSD

These are non-LSR VSAM files
- BUFNI = (#index_set_levels + 1) x #strings
- BUFND = 10 or (STRNO +1)

í DELETE/DEFINE the GCD and LCD for cold start

- may be faster than CICS cleanup processing
- BOTH GCD and LCD must be reinitialized together

í Use CEDA DEFINE LSRPOOL

For each LSR pool, specify in CSD: Maxkeylength, STrings, Data/Index BUFFER values.

Consider message DFHFC0208I 'LSR pool being built dynamically'. This is important if in total more than say 50 VSAM files use LSR.

- Reduces LSR pool build time during startup
- Reduces file OPEN times for first file in pool

Note that LSR pool must be dynamically built (first) in order to access the CSD file (which includes your definitions) for cold start, except an FCT would be used for the CSD.

CICS TS Partition Startup and Shutdown ...

Recommendations (cont'd)

Non-VSAM Hints:

- **Minimize sublibraries in LIBDEF search chain**

  - For faster startup
  - Especially when program loads are frequent

- **AUTOINSTALL terminals save initialization time**

  - Tune RAPOOL for optimum autoinstall rates
    (Fast autoinstall processing may need more RAPOOL than normal operation)

- **AUTOINSTALL programs also save initialization time**

- **Use the SVA for optional CICS modules in case of multiple CICS TS systems (and no CICS/VSE installed)**

  Preferrably use SVA-31 (for reentrant use of pgms)

  Use SVA-24 (apart from mandatory modules) only for modules to 'fill the gap by unavoidable 1M segment rounding'

  Refer to: 'VSE/ESA V2 Performance Considerations, Appendix A: Space Optimization'

- **Be aware of restrictions to load optional CICS modules into the SVA when CICS/VSE is also installed.**

  Refer to charts on 'CICS TS SVA Content'

CICS TS Partition Startup and Shutdown ...

CICS TS Startup Times

- **No systematic runs done so far. To be added**

- **Observed Startup Times**

  On our 9672-R11 with cached DASD: <1 min

  On a P/390 in our lab: 2 min 55 sec

- **AUTOINSTALL for programs can reduce startup times**

- **Hint for environments with very slow I/O devices**

  Very early tests with CICS TS on e.g. P/390 processors with very slow I/Os have shown the following problem/symptom:

  CICS TS startup time is slow due to many I/Os to program-load many phases.

  Since CICS TS requests to load about 300 phases in a single MVS LOAD request, that VSE system task is/was blocked (and thus not available e.g. for AR commands which also required a load, e.g. MAP, PRTY...)

  This problem of dominating the Program Load system task will be resolved.

Background Info

- **DL/I 1.11 is needed to work with CICS TS**
- **New functions and performance benefits**
  - **More 31-bit usage**

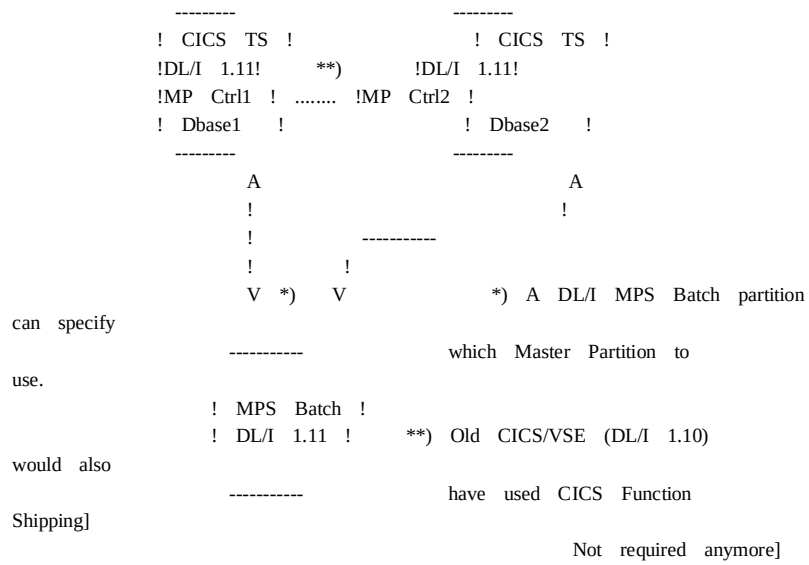
    Reduced 24-bit virtual storage constraints
  - **More, larger buffers in DL/I buffer pools**

    Reduced number of I/Os
  - **>4 GB DL/I data bases (up to 20 GB)**
  - **Multiple MPS control partitions**
  - **Enabled DL/I to participate in CICS TS SSP**

    DL/I system resources/areas in CICS-key protected storage

    Most simple multiple MPS configuration:

    2 DL/I data bases in 2 CICS TS partitions.
    (Each CICS has a started DL/I Master Partition Controller)



- **No Function Ship needed anymore, to go to a 2nd CICS partition**

CICS TS and DL/I VSE 1.11 ...

Changes vs CICS/VSE and DL/I 1.10

- **Multiple MPS control partitions**
- **>4 GB data bases: new in DL/I 1.11**
- More, larger buffers in DL/I buffer pools also in DL/I 1.10 with PTFs
- **Usage**

  DL/I DOS/VS 1.10 with CICS/VSE 2.3

  MPS only from old 1.10 DLZMPI00 to old DLZMPC00

  DL/I VSE 1.11 with CICS TS

  MPS only from new 1.11 DLZMPI00 to new DLZMPC00

  DL/I statistics setup as before, in PLTSD

Caution for environments with DL/I 1.10 and 1.11

- **Do NOT load any DL/I 1.11 phase into SVA, if also DL/I 1.10 is used**

  DL/I 1.10 would erroneously use them, be it in Batch or CICS/VSE

- **DL/I 1.10 phases could reside in SVA, if it's made sure that they are NOT used by DL/I 1.11:**

DL/I 1.11 Batch	DL/I 1.1[illegible] Batch is NOT allowed]
DL/I 1.11 CICS TS	DL/I SVA=NO in DLZACT req'd. (CICS TS SVA= w/o impact)
- New parameter SVA=NO!YES in DL/I 1.11 NUC generation for Online: DLZACT macro with TYPE=CONFIG. NO forces not to use any DL/I phase from SVA.	

Recommendations

- **Where possible, use new DL/I VSE 1.11**

  DL/I 1.11 does NOT run with CICS/VSE 2.3

Additional buffering

Also available in DL/I 1.10 with PTFs

Less 24-bit storage requirement

Also available in DL/1 1.10 with PTFs

You may divide DL/I data bases across CICS TS partitions

Also, more parallel concurrency for multiprocessors

Use >4 GB data base support, rather than increase application complexity

New support is transparent to application:

Only changes in DBDs. Needs a Reorg Unload/Reload and an ACBGEN

- **Loading of DL/I 1.11 modules into SVA**

  Not possible when also DL/I 1.10 is used, see sep. item)

  No DL/I phase needs to reside in the SVA

  With the exception of DLZACTL0 (used by LE), all DL/I 1.11 phases must reside below the line

  In total, 9 DLZ-modules are SVA-24 eligible, about 100K

  Use SVA-24, only when multiple partitions use DL/I 1.11.

PART F.

CICS TS Familiar Facilities

- Temporary Storage
- Transient Data
- Journaling
- Dynamic Transaction Backout
- Tracing

CICS TS Temporary Storage (TempStor)

Background Info

„ **A scratchpad facility to store data**

- Mostly short-to-medium-lived data
- Data can be shared between transactions

„ **Used within CICS itself, and by user applications**

- User pgms use it to accumulate data, or to pass data from task to task.
- Access is via queuename and position within queue.

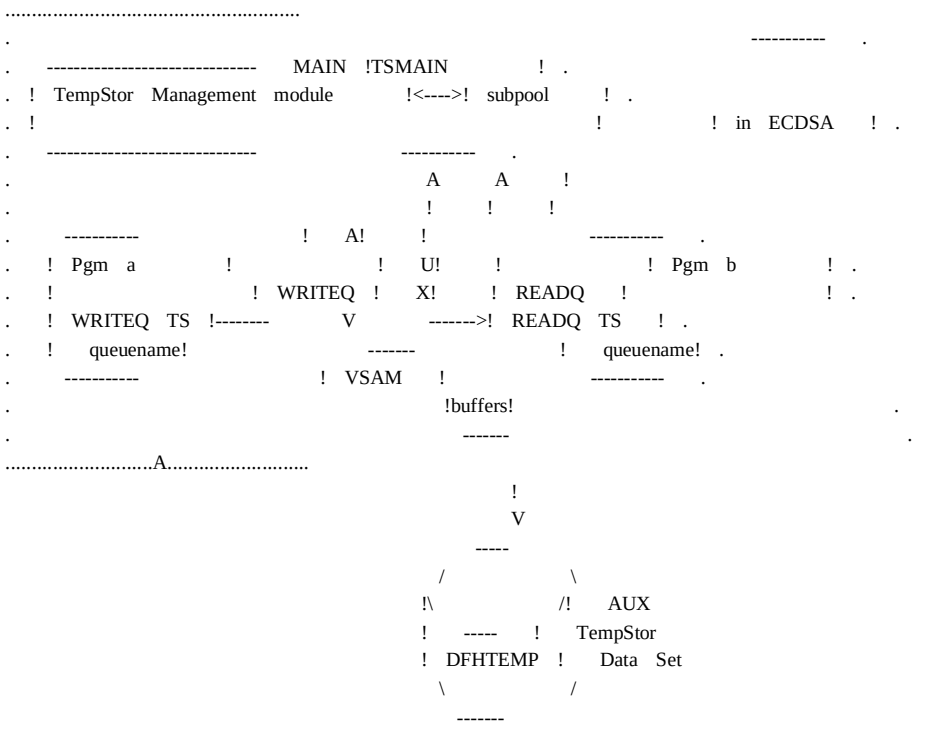
„ **Two types**

- MAIN TempStor

(in TSMAIN subpool of CICS ECDSA, volatile)

- AUXiliary TempStor

(in a VSAM ESDS 'DFHTEMP', accessed in CI-mode)



CICS TS Temporary Storage (TempStor) ...

Performance View

„ Requests are queued by TempStor queue name

which has pointers to its records.
Space allocated for pointers is controlled via SIT TSMGSET.
At any point-in-time, there may exist as many TSMGSET control blocks as currently required (means GETMAINs/FREEMAINs for new/empty TSMGSETs)

„ Physical I/Os only occur for AUX TempStor (DFHTEMP)

DFHTEMP is a VSAM ESDS file, used in CI-mode and User Buffering (UBF):

- With 1 CI per VSAM I/O
  - when a record is not in a buffer
  - when a CI is full
  - at the end of an LUW
    (only if recovery is set for the TS queue)
- Update WRITEs are possible anywhere (managed by CICS)
- READs can be concurrent
  (can exploit >1 extents for concurrent I/Os)
- The primary extent is formatted at CICS TS initialization,
  a secondary extent only when required
- A single CI can only contain data from 1 queue.

Changes vs CICS/VSE

MAIN TempStor area

Resided already above the line, but CICS/VSE was doing a lot of VSE GETVIS/FREEVIS requests.
This often resulted in longer GETVIS searches, especially when fragmentation was high. In extreme cases, MAIN TempStor needed more CPU time than AUX Temp Stor.
So, problems caused by VSE GETVIS-31 are gone

VSAM ESDS buffers

VSAM buffers for CICS TS reside in the TSBUFFRS storage subpool in ECDSA (31-bit).
They were BELOW the line in CICS/VSE, since DFHTSP was a 24-bit program using VSAM CI-mode processing

CICS TS Temporary Storage (TempStor) ...

Measurement Results

To be provided

Recommendations

- **Use MAIN TempStor for small, short-to-medium-lived data**

  if data need not be recoverable.

  By coding TS=(,0) in the SIT, MAIN TempStor is enforced

  No problems anymore with GETVIS fragmentation, since in a CICS DSA (ECDSA)

- **Check whether you can use MAIN TempStor instead of AUX TempStor**

  The maximum amount of virtual storage would be less than
  Max #CIs x CI-size
  since the CIs are not full.

- **Select a reasonably big SIT TSMGSET value (MAIN and AUX TempStor)**

  Refer to the SIT TSMGSET description.

  Value selection is even less critical than for CICS/VSE, since CICS GETMAIN/FREEMAIN is used.

- **Observe the CICS TempStor Statistics**

  For more specifics, refer to CICS TS Performance Guide.

CICS TS Temporary Storage (TempStor) ...

Recommendations for AUX TempStor

í For CICS/VSE AUX TempStor sometimes was beneficial to be put on VSE Virtual Disk

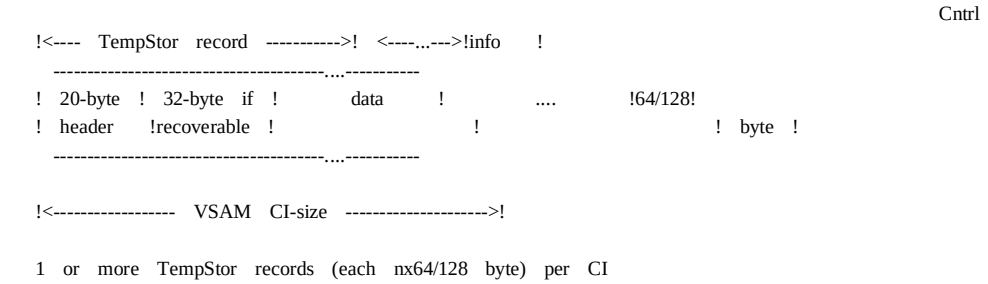
This was true in spite of additional code for doing virtual I/O, but only in case of very long GETVIS searches.

In CICS TS no GETVIS search problems occur anymore for MAIN TempStor.

í Use Recovery only when needed

Forces I/O at syncpoint

í Careful select the VSAM ESDS CI-size for DFHTEMP



Bigger CI-sizes are good, when access is mostly sequential (it is sequential within a single queue).
For more random access, bigger CIs may mean more data transferred than required, and CI-locking maybe more harmful to other requests.
Specify reasonably up to half the track size.

It's also wise to select the CI-size such that most of the TempStor records fit into a single CI, w/o the need of being spanned.

Cont'd

CICS TS Temporary Storage (TempStor) ...

Recommendations for AUX TempStor (cont'd)

- **Select a primary extent that mostly is big enough**

  Avoid the need for formatting a secondary extent on the fly

- **You may use >1 extent (on >1 logical volume)**

  To overlap physical READ I/Os

- **Usual file placement (volume selection) rules hold**

The next 2 items refer to SIT TS=(,bufnd,strno):

- **More TempStor buffers may reduce physical I/Os**

  Higher hit ratio when TempStor is doing lookaside in its buffers.
  3 buffers are the default.

- **Do NOT specify too few VSAM strings for DFHTEMP**

  - Allows maximum concurrency of I/Os
  - Is a small waste of storage if greater than #buffers
  - Cannot change serialization within a single TempStor queue

CICS TS Transient Data (TranData)

Background Info

„ Another means to hold (sequential) data: 'FIFO Queues'

- Processed sequentially (no random retrieval or update).
  Written and read only once (managed by 2 pointers to the oldest and newest data item in each queue): 'Queuing of data'

- Accessible and held in data queues
  ('destinations', defined in DCTs)

- Shareable between transactions

Two forms exist:

„ INTRApartition TranData

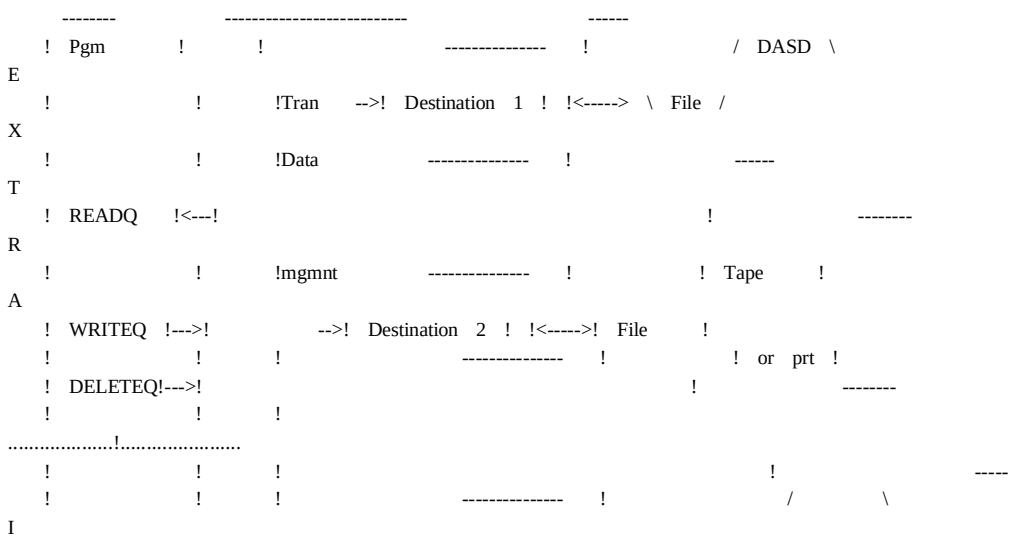
(within 1 CICS region)

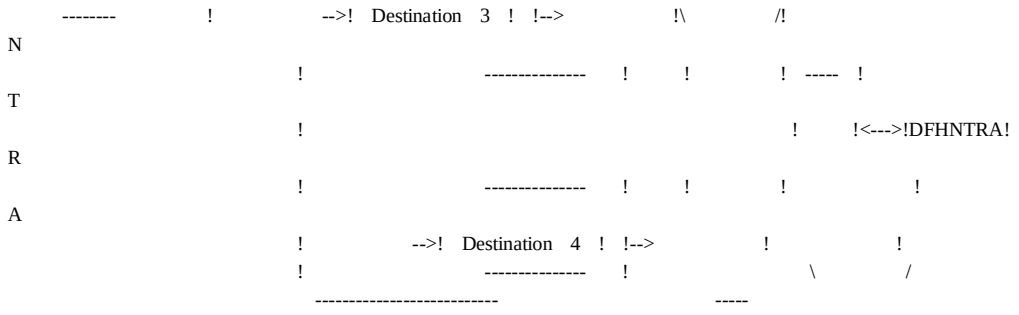
Data are stored in the CICS INTRApartition data set CICS.TD.INTRA
(a VSAM ESDS file, usually defined with REUSE=YES, i.e. when no need exists to process the data set after CICS shutdown)

„ EXTRApartition TranData

(access also from outside, e.g. batch)

Data are stored on DASD (DTFSD) or Tape (in/output) or sent to printers (output)





Destination 3
Destination 4
N
T
R
A
DFHNTRA

CICS TS Transient Data (TranData) ...

„ TranData used by application pgms

- to queue data
- to initiate other tasks
- to transfer data between tasks
- to capture data for batch processing

Performance View

Requests are queued by TranData 'destination'

INTRA:

VSAM processing is 'ESDS in CI-mode with User Buffering'

VSAM buffers reside above the line in ECDSA

1 VSAM CI can only hold data for 1 destination

No spanned records supported

Default is 3 buffers and 3 strings

EXTRA:

SAM GETs and PUTs are used for the BAM file,
with imbedded WAIT logic (which may cause CICS to wait)

Changes vs CICS/VSE

VSAM buffers are in ECDSA now, formerly in 24-bit, since DFHTDP was a 24 bit program using VSAM CI-mode processing.

CICS TS Transient Data (TranData) ...

Measurement Results

Early runs have been done. Final data will be provided.

Recommendations for TranData

INTRA and EXTRA:

- **Observe the CICS Transient Data Statistics**
- **Use Recovery for TranData only when needed**

INTRA:

- **Refer to recommendations for CICS AUX TempStor**
  - Size of primary extent
  - Min. CI-size (should be longest record + 32 byte, max is 32K)
  - Number of VSAM buffers and strings
- **Use the shipped SIT TD=(bufnd,strno) values, increase when TranData INTRA statistics suggest.**

EXTRA:

- **Do not specify a physical printer**

  Use a logical printer, preferrably a POWER 3800 printer
- **Use BUFNO=2, and BLKSIZE of about 2 per track**

  in BAM file definition.
- **You may consider a redesign of the application to use native VSAM ESDS instead of TranData**

  Also provides recovery of these data

Background Info

- **Collect information about activity carried out by your CICS system ('Journaling', 'Logging')**

  - Before- and after-images of recoverable resources (prime use)
  - Info for auditing or accounting (other uses)

  Required for CICS Recovery

- **Activity Keypoint data are written to the CICS System log**

  - Snapshots of system activity (info on in-flight tasks)
  - Frequency determined by SIT AKPFREQ parameter

- **Journals are on disk ('sequential file') or tape**

  Disk journals must be in BAM space,
  no VSAM managed SAM, no VSAM ESDS]

  System log or user journals, defined in a JCT

  Optionally, a user journal can also be written to DMF.

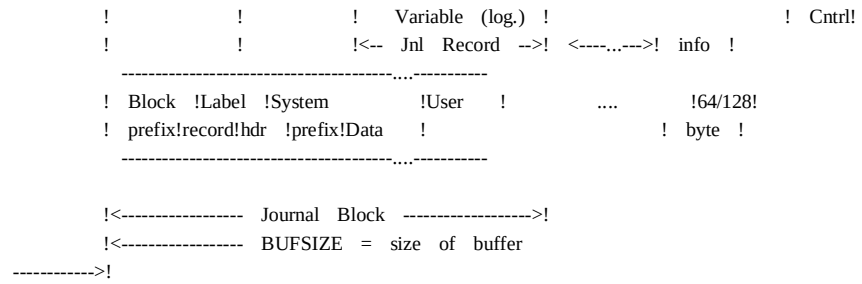
  CICS wraps data around if a journal data set is full,
  but often an A and B data set are used alternately

- **Switching between 2 files can be done automatically**

  (e.g. between CICS System Log DFHJ01A and DFHJ01B)

Background Info (cont'd)

„ Journal Buffer



-> 1 or more log. Jnl Records in a single phys. Journal Block

-> Jnl Blocks are variable, max. size is BUFSIZE

Performance View

„ Journaling is done by separate subtasks

No performance benefit for VSE

„ BUFSIZE in JCT

- is the size of each of the 2 buffers per journal
  (located in DSA below the line)

and

- is also the maximum physical record (block) written

„ Data (journal blocks) are written to the medium

- when current buffer is full or
- when ICV has elapsed since the first jnl record entry
- immediately (i.e. 'forced WRITEs'), see table below,
  together with data accumulated so far in that buffer

WRITEs to DASD are Format-WRITEs (on top of pre-formatting)

CICS TS Journaling ...

Changes vs CICS/VSE

„ **1 buffer per journal was replaced by 2 buffers (each of BUFSIZE) in CICS TS:**

CICS TS uses 1 buffer for output to the device, while receiving input from transactions into the other buffer

- **No BUFSUV (buffer shift-up value) required anymore.**
- **For same BUFSIZE, twice the space is required.**

Journal I/O Initiation and Recommendation

Transactions initiating jnl requests only can proceed, when journaling data are on the physical medium.

	SYSWAIT=STARTIO (default)	SYSWAIT=ASIS
Synchronizing jnl req. from CICS mgmnt modules	immed. As in STARTIO of the request	
All other jnl requests	As in STARTIO As in STARTIO	
Jnl request have STARTIO=YES!NO (immed!delayed), usually it is NO. So, SYSWAIT=ASIS usually causes delayed WRITEs for jnl requests from CICS mgmnt modules		

- **Use SYSWAIT=STARTIO (default) for all journals to obtain best transaction response times**
- **SYSWAIT=ASIS should be avoided whenever possible**

If at all, ...
- should be only used for user journals and
- only when journal device is overloaded and
- only when multiple tasks use the same journal

CICS TS Journaling ...

Measurement Results

Case JA	AKPFR	CPU util	NPS	txn/sec	Av. RT (sec)	CPU-time /txn(ms)	ITR-ratio	
NO	(0)	36%	0.290	20.33	0.073	17.92	1.00	
24	0	41%	0.326	20.32	0.093	20.34	0.88	
24	200	41%	0.327	20.31	0.093	20.44	0.877	

- Standard system setup and environment
- IO/txn increased from 4.35 to 5.45 (no keypointing)
  and to 5.55 (incl. very frequent keypoints 1 in 8 sec)

+12% more CPU-time per txn by Journaling for DSW

Keyointing, though frequent here, with very small impact

More Recommendations

í Observe the CICS Journal Control Statistics gathered for each journal

- number of jnl requests
- number of I/Os (blocks written)
- avg length of blocks written
- number of times buffer was full

Distinguish between the CICS System log and user journals
(System log has a higher share of SYNCHRONOUS requests)

í Select a reasonable BUFSIZE in DFHJCT

- At least as large to contain the biggest journal record
  (min= 512 byte)

- At most as large as tracksize (max= 2760 byte)

- If frequency of buffer full situations is too high,
  (i.e the receiving buffer has filled before the I/O on the alternate buffer was completed) ...

  increase BUFSIZE (system and user journals)

- Synchronous jnl requests cause avg blocksize to be
  smaller than BUFSIZE

CICS TS Dynamic Transaction Backout

Backgound Info

- **Changes to recoverable resources are logged**
  - to 1 dynamic log for each task
  - to the CICS System log (used for restart)
- **Required to un-do changes by**
  - **abended txns**
  - **txns in flight when CICS abended**
- **DTB is always available, when Journaling is active**

Performance View

- **DBUFSZ in SIT is the max. size of the dynamic log buffer (1 per active task)**

  Can spill to TempStor.

Recommendations

- **Observe the Dynamic Txn Backout Statistics**
- **Refer to the SIT DBUFSZ recommendations**

CICS TS Tracing

Backgound Info

- **Trace the CICS requests issued by CICS applications**

  Collect valuable information for problem determination, and in specific cases, for a more detailed performance insight.

- **Two types exist:**

  ### - Internal Trace

  Types of trace entries written to the internal trace table:

  - CICS system trace (when SYSTR=ON, default)
  - User trace (when USERTR=ON, default)
  - Exception trace (always active)

  ### - AUXtrace

  Tracing into CICS AUXiliary data set

Performance View

- **Considerable overhead in CPU-time, plus I/O (AUXtrace)**

  Overhead depends on frequency of CICS services called

- **Trace entries vary in size (4K max)**

- **Trace table is always allocated, and above the line**

  Minimum and default size is 16K, bigger values needed

- **AUXtrace uses 2 buffers (4K each) in GETVIS-24**

  Outside DSAs

- **4K data per AUXtrace I/O**

- **Trace code now resides above the line**

CICS TS Tracing ...

Changes vs CICS/VSE

- **Trace usage is much more flexible regarding**
  - **Selection of trace type**
  - **Control via new CETR txn (full screen, easy) and SET TRACEFLAG command**
  - **Switching of trace extents**
  - **Processing of trace (new utility DFHTU410)**
- **More trace points, much more info per entry**

Measurement Results

CICS Internal Trace CPU-time cost		
CICS/VSE	DSW RAMP-C	+18% + 8%
CICS TS	DSW	+18%
- Note, DSW is very CICS function intensive		

Recommendations

- **Select a reasonable size Internal Trace Table**

  The SIT TRTABSZ minimum is 16K, VSE/ESA 2.4 ships 80K.
  Use at least 5 times the size from CICS/VSE

- **Swich Internal Trace OFF, when your CICS production environment is stable**

  I.e. no new applications installed, system changes (if any) also OK.

  Exception Trace entries still are made.

- **Select larger AUXtrace data sets**

CICS TS AUXtrace Usage

- **For vital and typical transactions:**

  Save CICS AUXtraces before you migrate

  May help in case of unexpected performance problems

AUXtrace Function

- **AUXtrace provides the most detailed level of data for performance monitoring/tracing**

  Typical usage scenarios are the analysis of

  1. a specific transaction in detail
     - which requests are issued, to which resources, ...

  2. resource contention between different tasks
     - on which resource a task waits, what task holds the resource

AUXtrace Activation/Control

- **You may now use CETR full-screen I/F to control CICS tracing activity (Internal and AUXtrace)**

  TS AUXtrace entries are only collected when Internal trace is active]

  - Internal Trace can be activated via SIT INTTR=ON,
  - AUXtrace can be activated via SIT AUXTR=ON,

  but both ways may not be flexible enough.

- **Or, better, use CEMT SET AUXtrace STArt or STOp**

- **Only have AUXtrace active for a small interval**

  This may be a production interval, or a single test txn only. Individual transactions can be traced also via the TRACE keyword in the TRANSACTION definition.

CICS TS AUXtrace Usage ...

CICS AUXtrace Activation for 'Performance'

Tracing a single transaction, to see CICS functions used and timings

Ù General Rules

- **Make sure (best case) that no other VSE activity is running**

  (in order to get repeatability and comparability)

- **Make sure only the CICS activity to be traced is done**

- **Make sure that all files accessed are OPEN**

  Avoid that CICS must open them under the cover, resulting in even more trace entries. Use CEMT SET FILE OPEN if required.

- **Be prepared for lots of output, even when tracing few seconds only**

Ù Sequence of Actions

In CICS TS you can use CETR full screen, or as shown here, CEMT.
To be done in any affected CICS partition.

CICS TS	CICS/VSE
CEMT SET INTtrace STArt	CEMT SET TRACE ON
CEMT SET AUXtrace STArt	CEMT SET AUXtrace ON
\\\\\\\\ measured activity	\\\\\\\\ measured activity
CEMT SET AUXtrace STOp	CEMT SET AUXtrace OFF CLOse
CEMT SET INTtrace STOp	CEMT SET TRACE OFF

CICS TS AUXtrace Usage ...

AUXtrace Processing

Use the provided job skeletons from ICCF Lib 59

Check LIBDEFs and DLBLs used in these jobs before submitting them.

- **Member DFHAUXOL (for CICS/VSE, program DFHTUP)**

- **Member DFHAUXPR (for CICS TS, program DFHTU410)**

  This Trace utility program provides capability to selectively print entries by:

  TASKID!TERMID!TRANID!TIMERG!TYPETR

  For CICS TS performance investigations, do not forget to use the parameter settings for getting timestamps:

DFHPD410: CALL DFHPD140 DATA TRS=<TIMESTAMP,INTERVAL=1.0>
DFHTU410: TIMESTAMP,INTERVAL=1.0 as a parameter

AUXtrace Comparison:

- **For pure problem determination (detailed level), AUXtrace in CICS TS is much better than in CICS/VSE**

- **For customer fast usage, AUXtrace in CICS TS is less readable**

- **Performance investigations with AUXtrace may require multiple (print and analysis) iterations**

CICS Partitions and IUI/ICCF

Some Known Facts on IUI/ICCF

- **IUI still is very valuable for ad-hoc system programming tasks**
- **Most CICS based IUI functions do not require ICCF**

  Some type of functions that need ICCF:

  - calling a program in an ICCF I/A partition
  - using ICCF forground functions (e.g. Edit)

- **ICCF so far used the CICS Macro-level**

News on ICCF in VSE/ESA 2.4

- **Upgraded to use the CICS Command-level**
- **No more CICS control blocks are directly accessed**
- **More IUI programs were made 31-bit eligible and now may reside above the line**

  More info will be provided

- **Compiled against CICS TS**
- **ICCF in VSE/ESA runs only together with CICS TS**

  Also note that only 1 ICCF is allowed in any VSE system (Don't try to 'migrate' old ICCF from VSE/ESA 2.3, refer to Information APAR II11987)

CICS Partitions and IUI/ICCF ...

Survey on CICS Partitions in VSE/ESA 2.4

CICS Release 'APPLID' jobname partition	Installation and Startup	ICCF and IUI
CICS TS 'DBDCCICS' CICSICCF F2	Fully installed, automatically started/active	All IUI and ICCF functions
CICS TS 'PRODCICS' PRODCICS F8	Only startup skeleton (no ICCF), not started by deflt	Only IUI functions available which do not need ICCF
CICS/VSE 'OLDCICS' OLDCICS F4	Install from Extended Base Tape req'd. Only startup skeleton provided	No IUI at all, but common Signon security available via exit DFHXSE

- Names and partitions shown are shipped values
- Here only Environment B is considered

CICS TS and TCP/IP for VSE/ESA

TCP/IP can also be used to communicate with CICS TS

This will not make VTAM obsolete, as it neither does with CICS/VSE

- **Most TCP/IP functions do not have direct I/F to any CICS**

  E.g. TN3270 is a VTAM application

- **CICS clients (for Telnet, FTP, LPR) do also work on CICS TS**

- **TCP/IP APIs likewise hold for CICS TS**

- **Any performance aspects and data apply shown in**

  'IBM VSE/ESA TCP/IP Performance Considerations'

PART G.

CICS Intercommunication

This part will be extended.

So far, main emphasis was dedicated to Shared Data Tables, which allow to avoid MRO FS for 'remote READs'.

Essentially, CICS MRO performance is expected to be similar to CICS/VSE, if in CICS/VSE all available PTFs are used.

CICS Intercommunication

- **CICS Interregion Communication (IRC)**

  Communication between CICS regions in the same operating system.

  Communication is done via MRO (Multi Region Operation)

- **CICS Intersystem Communication (ISC)**

  Communication between CICS systems (and others) in different operating system images

  Communication is done via VTAM SNA.

- **MRO is more efficient than ISC**

  - shorter pathlength
  - fewer components involved
  - no line delays

Emphasis here is laid on MRO

CICS TR and FS

(Mostly used) CICS Intercommunication Functions

Ù Transaction Routing (TR)

A terminal owned by CICS A can run a txn owned by CICS B

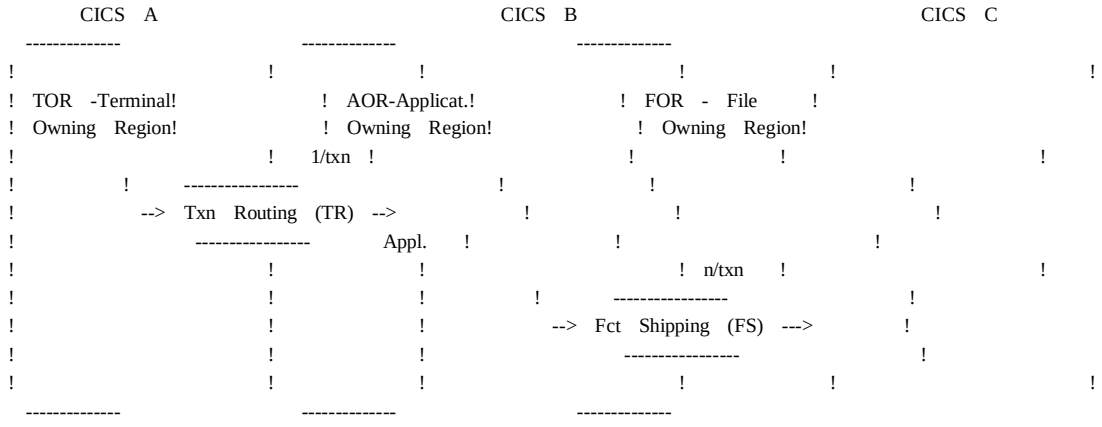
Ù Function Shipping (FS)

Access to a remote file, DL/I database, TranData queue, TempStor queue

Multi-Region Configuration

All 3 CICS systems may reside

- in the same VSE/ESA system (IRC/MRO)
- in different VSE/ESA systems (ISC)



CICS TS and MRO Functions

„ MRO was extended to allow CICS TS to communicate with CICS/VSE

CICS TS ships a new Interregion Communication Program DFHIRP

Transaction Routing (TR)
Function Shipping (FS)

(CICS TS Shared Data Table Support is an 'enhancement' to FS)

Distributed Transaction Processing (DTP)

Requires a specific application design to exploit DTP

Distributed Program Link (DPL)

Refer to the CICS TS Intercommunication Guide, SC33-1665

„ Allows to use withdrawn facilities for activities initiated in a CICS TS partition (during CICS migration period)

„ CICS TS provides more MRO/ISC control

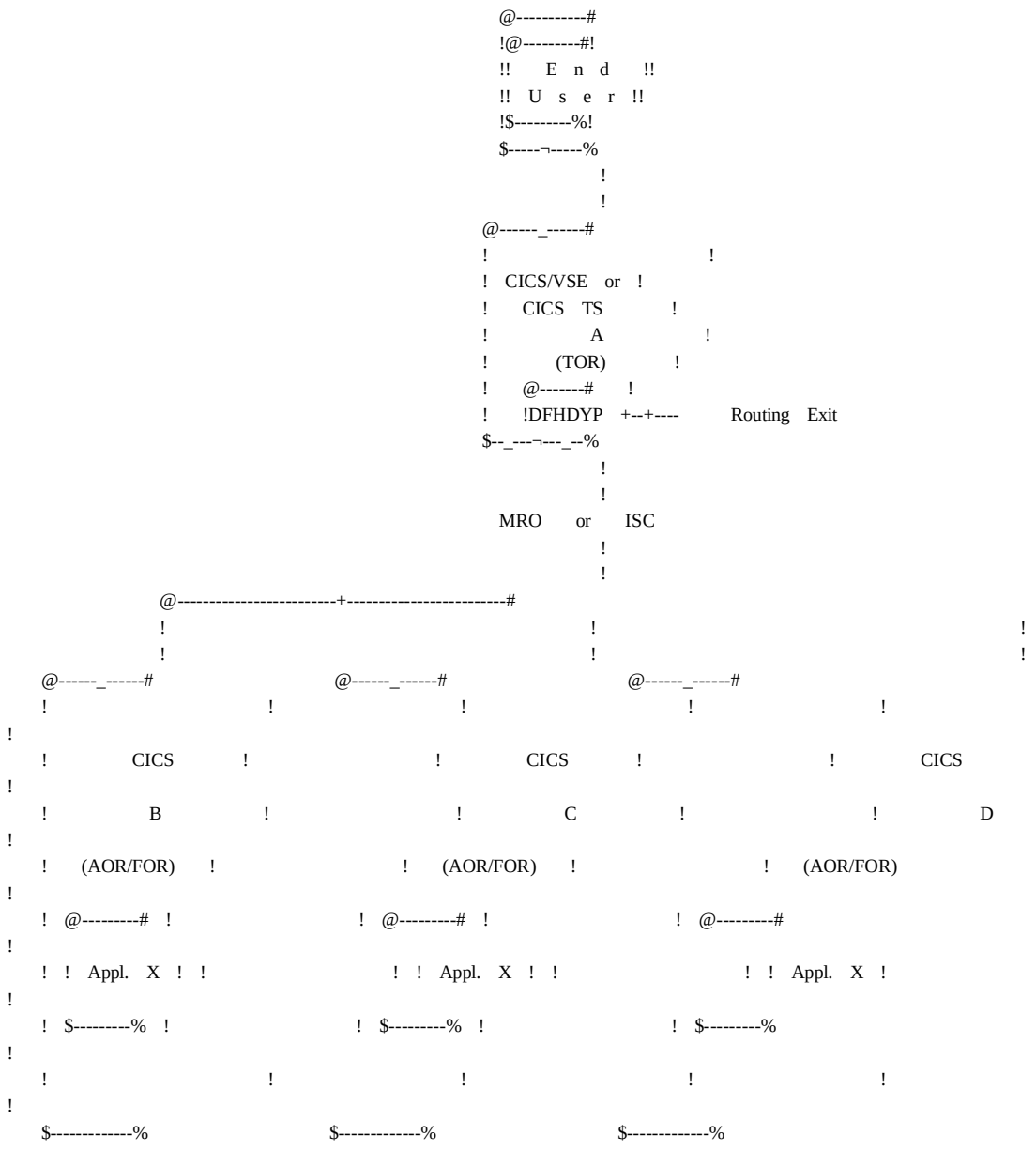
- QUEUELIMIT and MAXQTIME in CEDA DEFINE CONNECTION
- XZIQUE exit

(besides NOQUEUE!NOSUSPEND) in EXEC CICS ALLOCATE for DTP, already available for CICS/VSE)

„ Care needed with SVA modules

Refer to 'CICS TS SVA Content' foils

Dynamic Transaction Routing



Dynamic Routing could be done e.g.

- cyclically
- random
- load dependent
- based on key ranges

Performance Hints for CICS MRO

- **Carefully select the subdivision of CICS files and applications into different/same CICS partition(s).**

  - For TR only 1 MRO request is done per txn
  - For FS commonly many MRO requests are needed per txn

- **If possible, avoid too frequent FS requests**

  - per transaction
  - in total

  CICS TR is less critical regarding CPU-time overhead.

- **Check your MROBTCH setting**

  If >1, the target CICS is not posted until this number of MRO events occur before the target CICS is posted. In any case, after the ICV interval, requests are forwarded.

  By batching the MRO request, the CPU-time overhead of dispatching the target CICS may be reduced.

  Usually, it is recommended to use the MROBTCH default of 1, since the saving of CPU time is not so big in order to compensate for the MRO delay.

  CEMT or EXEC CICS SET SYSTEM MROBATCH(value)

  can be used to dynamically change the value and find the best compromise for your system.

  In any case, avoid a higher MROBTCH value together with a high ICV value in systems with low MRO activity.

Performance Hints for CICS MRO (cont'd)

í Always use MROLRM=YES

Reduces MRO CPU overhead by keeping the mirror transaction in the target CICS longer active, in expectation of additional FS requests from the same transaction in the originating CICS.
Be careful for dialog oriented (conversational) transactions.

Note that the CICS SIT default is MROLRM=NO, newer VSE/ESA setups for CICS TS now already contain MROLRM=YES.

í ISC=YES is required to do any MRO (or ISC) functions

ISC=YES,NOFS from CICS/VSE 2.3 does no more exist, since it never was required on CICS/ESA.

Reason: - With CICS TS, the check whether a file is remote or not is done efficiently.

í Check the number of available MRO/ISC communication sessions

RECEIVECNT and SENDCNT in DEFINE SESSION

Values should be ...

- big enough that, if at all, only small queueing for communication sessions occurs.
  Compare to figures from CICS Terminal Statistics

- not too large to avoid waste of 24-bit virtual storage
  - about 40 byte per connection and per defined session, plus 16 byte per opened session

í Observe CICS Statistics from DFHSTUP

'ISC/IRC SYSTEM AND MODE ENTRIES'
- File control FS requests
- TD FS requests and TS FS requests
- ...

í FS can be favored in partitions,

when higher transaction priority is assigned to CSMI, CSM1, CSM2, CSM3 and CSM5
Default priority is low, you may select 255

CICS TS External CICS I/F (EXCI)

ù New Function

Allows a batch pgm to call a CICS application online

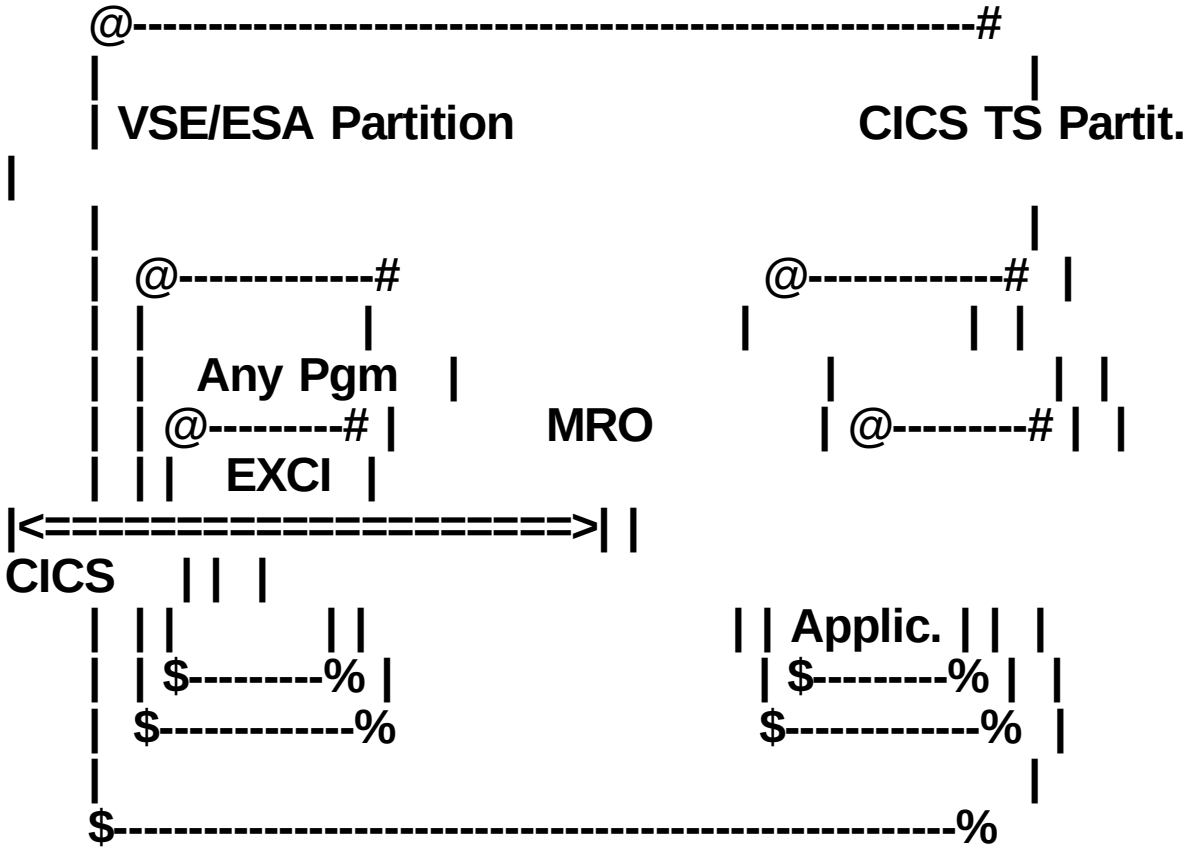
An API, allowing a non-CICS program (e.g. batch 'client') to

- call a program running under CICS TS ('server')
- pass and receive data by means of a communication area

The CICS TS program is invoked as if it were linked by another CICS program.

EXCI is the S/390 equivalent of the ECI (External Call Interface) function provided by CICS clients.

DFHIRP will be used, as for usual MRO connections.



Ù New Operational Possibilities

Could be used e.g. to close/reopen a CICS file, or to disable/re-enable transactions during the run-time of a certain Batch program (e.g. backup and recovery)

Two types of EXCI related calls are possible:

- EXEC CICS LINK PROGRAM (1 communication each)
- EXCI CALL (multiple communications possible)

EXCI needs SIT IRCSTRT=YES, together with DEFINE CONNECTION, SESSION and PROGRAM (the server pgm in CICS TS)

PART H.

CICS TS System Initialization Table

- **Many SIT parameter differences**

  (about 90 new, 30 changed, 30 obsolete)

  The following shows potentially performance relevant CICS TS SIT parameters in (nearly) alphabetical order.

- **VSE/ESA supplied SIT for CICS TS provides functionality and defaults similar to CICS/VSE 2.3**

  - **BUT careful review is essential**

SIT AKPFREQ

AKPFREQ

AKPFREQ=200	Activity keypoint frequency

Specifies the number of consecutive blocks written to the CICS System log, that triggers the activity keypointing, i.e. to also write a snapshot of the system activity to the CICS System log (this includes scanning of many CICS control blocks).

A higher value of AKPFREQ
- reduces the amount of data on the System log, though slightly more data are written per keypoint
- increases emergency restart times

A lower value of AKPFREQ
- increases the amount of data on the System log
- reduces emergency restart times (by limiting the backward scan of the System log)

0 means no keypointing (i.e. no emergency restart possible).
200 is the SIT default and the shipped default for VSE/ESA.

AKPFREQ is not applicable when no CICS System log exists (e.g. when no actual Journaling is done due to JCT=NO)

í Set AKPFREQ such that about every 15 minutes a keypoint is written

Use the CICS transaction statistics to see the frequency of execution of the CSKP transaction.

SIT AUXTR ...

AUXTR

AUXTR=OFF	Auxiliary trace control

This parameter controls whether trace entries are written to the AUXtrace data set.
AUXtrace entries are only possible, when Internal Trace is also started.

OFF is the SIT AUXTR default and the shipped default for VSE/ESA.

- **Use the default, and activate only temporarily via CEMT SET AUXTR or via CETR**

CSD ...

CSDLSRNO=1	CSD LSR pool number
CSDSTRNO=4	CSD string number

The CICS System Definition file (CSD) must now be defined in the SIT.

It is recommended to use VSAM LSR for the CSD KSDS file

- **Keep the CSD as VSAM LSR file**

- **Use CSDSTRNO >2 for concurrent CSD access (2 strings per user).**
  Shipped default of 4 is enough, usually.

SIT DBUFSZ ...

DBUFSZ, DBP

DBUFSZ=2000	Max. size of a dynamic log buffer
DBP=1$	Dynamic Transaction Backout pgm

Maximum size of the dynamic log buffer for each active txn, used to store a copy of the backout records.

Initially, CICS uses half that size. Also space exists for spilling.

All these areas reside above the line (EDSA), thus the VSE shipped default of 2000 byte is bigger than the SIT default of 500.

- **Increase DBUFSZ if the '# records spilled' is consistently over 2% of the '# records logged'**

- **Any choice of DBP is OK, as long as function is OK**

  Since the DBP program resides above the line, the somewhat bigger 2$ program (also includes functions for DL/I) would not hurt as much as it would have done before.

  Refer to the Dynamic Transaction Backout Statistics

SIT DSA values

DSALIM, EDSALIM

DSALIM=5M	Max. total size of the 4 DSAs below the line
EDSALIM=25M	Max. total size of the 4 DSAs above the line

These parameters specify the upper total size of the area for the DSAs below and above the 16M line:

	Minimum SIT	default Increment	VSE shipped	
DSALIM	2M	5M	256K	5M
EDSALIM	10M	20M	1M	25M (Env B)

- **Recommendation**

  Refer to the CICS Statistics 'Storage manager: global statistics' showing
  Current (E)DSA limit/total/peak and more

xDSASZE, ExDSASZE (SIT overrides only)

xDSASZE=0K!nnnnK	Size of xDSA or ExDSA (SIT overrides only)
ExDSASZE=	x=C,S,U,R

These parameters allow to specify fixed size xDSAs and ExDSAs.

0K is the default, and means a dynamically changing size.
Any other value is fixed (and rounded to multiples of 256K or 1M).

You can specify 1 (or more) DSAs with fixed size, e.g. for

- ERDSA to ensure, reentrant programs do NOT swamp the ERDSA
- ECDSA to ensure, always storage for TSMAIN is available

In case 3 DSAs (below or above) are 'fixed', the remaining DSA is also fixed, and only varies with DSALIM or EDSALIM.

- **To start, let CICS dynamically adjust the sizes.**

  Fix the size of 1 or more DSAs only in specific cases

ICV, ICVR, ICVTSD

ICV=1000,	Interval control value in msec
ICVR=20000,	Runaway task interval
ICVTSD=200,	Terminal scan delay

ICV is the maximum time a CICS gives control away. Basicly it is used to avoid CICS modules being paged out during periods of low activity.

So it's rarely used in busy production systems, and hardly required in systems with adequate real storage.

ICVR is the time after a task is considered to be in a CPU loop.

ICVTSD is an interval to control dispatching of terminals.

í Leave ICV=1000

ICV value selection is not critical in general.

Use only a smaller value (100 or 200 msec), if, together with MROBTCH>1, you get MRO benefits.
A too high ICV may negatively affect response times for CICS shutdown, MRO Batching ...

í Let ICVR as is

Monitor AICA abends. Value is independent of ICV

í Set ICVTSD to a value smaller than ICV, use ICVTSD=200

Though CICS TS is VTAM only, and only BTAM caused long TCT scans...

ICVTSD=0 may negatively impact I/Os to non-VTAM devices with WAIT (sequential terminals or consoles) and ATI requests.

Measurement Examples:

For a single DSW partition (no MRO used) and about 36% CPU utilization <0.5% CPU and RT variation was observed for all of the following cases

ICV=	ICVTSD=200 (Base)
100000	200
1000	0
100	100

SIT INTTR ...

INTTR

INTTR=ON!OFF	Internal trace control

This parameter specifies whether the any of the 3 types of CICS trace entries are written to the internal trace table.

INTTR=ON is the default.

Related parameters are SYSTR and USERTR, which are recommended to be ON (= enabled).

- **You may consider switching internal tracing off, as soon as the CICS partition is stable, and only well tested new applications are added**

  To temporarily activate internal trace, use CEMT or CETR.

IRCSTRT, ISC

IRCSTRT=NO	Interregion communication start
ISC=NO!YES	Intersystem communication control

IRC must be started before any MRO can be used.
If used occasionally, CEMT SET IRC OPEN is available.

ISC=YES

- is required for any MRO or ISC usage (to load programs).
- avoids communication errors and the need for operator intervention.
- does NOT give CPU-time overhead (as in CICS/VSE) (No need or means in CICS TS to specify ISC=NO,NOFS).

SIT JCT ...

JCT

JCT=NO	Journaling control

Journaling (system and/or user journals) is only done if a JCT is being used, specified via YES or suffix.

Measurement Example:

A single DSW partition with 1x300 user:

+12% CPU-time increase with Journaling

(Refer to CICS TS Journaling foils)

MN ...

MN=OFF	Monitoring control
MNCONV=NO	Conversational task monitoring control
MNEXC=OFF	Exception class monitoring control
MNEVE=OFF	Event monitoring control
MNPER=OFF	Performance class monitoring control
MNSYNC=NO	Syncpoint monitoring control
MNFREQ=0	Monitoring frequency

These parameters refer to CICS Monitoring.

The settings shown above are the SIT defaults and also the values shipped by VSE/ESA for the CICS TS.

- **Do not change parameters, unless monitoring req'd**

  (maybe for a CICS TS performance monitor)

MN=ON, together with MNCONV=YES, MNEXC=ON, MNFREQ=10000, MNPER=ON, MNSYNC=YES and DMF:

+11% of CPU-time increase vs MN=OFF

MRO ...

MROBTCH=1	MRO batching value
MROLRM=YES	MRO long-running mirror task option

These specifications allow to reduce MRO overhead by

- clustering TR and FS requests (MROBTCH)

- keeping the mirror transaction always active, expecting additional FS requests from the same originating transaction (MROLRM)

í Leave MROBTCH=1 (default)

Usually, the savings in CPU-time do not outweigh the negative impact on response time.

Only at very high utilizations with heavy MRO, MROBTCH>1 may be beneficial, but only together with not too high ICV.

MROBTCH can be overwritten via CEMT Set SYStem MRobatch(value)

í MROLRM=YES is recommended in general

SIT MXT

MXT

MXT=80	Max number of concurrent tasks

Specifies the maximum number of user(]) tasks, CICS TS allows to exist at any point in time.

It primarily controls the total CICS system virtual storage use, especially to avoid overcommitted resources such as storage (SOS) and others.

For this amount of concurrent tasks, the control blocks are pre-allocated in CDSA and ECDSA.

MXT can be altered via CEMT SET MAxtasks(value)

AMXT and CMXT parameters (Average and per Class) no more required.

- **Monitor the CICS TS Statistics 'Number of times the MXT ceiling has been reached'**

- **Select an MXT value**

  - not too low (avoid unnecessary delays)

  - not too big (wastes storage, is new)

- **It is NOT reasonable to set MXT to 999**

  This was often done in CICS/VSE, in CICS TS this task storage is pre-allocated.

  For consequences of too high MXT value (VS constraints), refer to the following measured results.

MXT (cont'd)

GETVIS and DSA sizes observed (MXT variation)

		MXT=80	MXT=999	DELTA 919 tasks per task
GETVIS-24	SIZE	11260K	11260K	0 -
	USED	7688K	8068K] 380K	0.4K
	HWM	11260K	11260K	0 -
CDSA	Curr.Size	512K	2816K] 2304K	-
	Curr.Used	416K	2612K] 2196K	2.4K
	Peak.Used	436K	2632K] 2196K	2.4K
	Curr.#ext.	2	11 9	
	#GETMAINs	357	1363 1006	1.1
GETVIS-31	SIZE	29696K	29696K	0 -
	USED	29656K	29660K 4K	0
	HWM	29696K	29696K	0 -
ECDSA	Curr.Size	3072K	15360K] 12288K	13.4K
	Curr.Used	2812K	15124K] 12312K	13.4K
	Peak.Used	2812K	15124K] 12312K	13.4K
	Curr.#ext.	3	15 12	
	#GETMAINs	12031	13961 1930	2.1

- A GETVIS-24 requirement of about 400 byte/task (why??)

- **No delta in total GETVIS-31 sizes, BUT deltas in CDSA/ECDSA**

  This indicates that CICS TS GETVISes DSALIM and EDSALIM from scratch (i.e. BEFORE a need for a DSA extension arises).

  2.4K and 13.4K per task (in CDSA and ECDSA) fits nicely to 2K and 12K per task plus 10% expansion stacks.

PRTYAGE

PRTYAGE=5000	Priority aging interval

This parameter was introduced to allow low-priority txns to have some chance to proceed in a system which is dominated by high-priority transactions.

This parameter specifies the time interval in msec used in the priority aging algorithm for incrementing the priority of tasks.
Each time a task is in the ready queue and not dispatched for PRTYAGE msec, priority is increased by 1.
After an 'aged' tasks is dispatched, its priority is returned to the original value.

A value of 0 means no priority aging, the SIT default is 32768, the shipped VSE default is 5000 msec = 5 sec.

- **Use prioritization sparingly**

  By rotating tasks in the ready queue, some fairer share should apply.

  Refer to the CICS TS Performance Guide

RUWAPOOL

RUWAPOOL=NO!YES	LE/VSE run unit work area pool

This parameter allows LE/VSE to use a pool of work areas for run unit control blocks (YES) instead of releasing the area after each exit of an LE run time unit (enclave).

SIT default is NO, YES is recommended in general

SIT RAMAX ...

RAMAX, RAPOOL

RAMAX=256	Size of VTAM RECEIVE ANY input buffers
RAPOOL=10	Number of concurrent RECEIVE ANY requests

RAMAX and RAPOOL refer to VTAM RECEIVE-ANY operations:

- Size and number of RECEIVE-ANY input areas (RAIAs) for VTAM input messages, residing in the fixed pageable subpool below the line.

Appropriate values allow VTAM to put all the input messages directly into CICS buffers. An RAIA is available for reuse by another input message as soon as CICS has moved the data into a TIOA.

- **RAMAX should be ...**

  - **not too small, so CICS and VTAM will not split an incoming message into pieces**

  - **at least as big as the largest CEDA RECEIVESIZE for any frequently-used terminals**

  - **set such that it accomodates a high share of the incoming requests**

- **RAMAX default of 256 byte is usually appropriate**

  Maximum inbound SNA 3270 RUSIZE is 256 byte

- **RAPOOL should be such that VTAM queuing activity is avoided for all except real peak traffic**

  As a starting point, you may use a RAPOOL value of 1.5 times the peak number of messages per second.

  'Number of times reached maximum' < 5% of input messages

  Observe in the CICS VTAM Statistics
  'Maximum number of RPLs posted'
  'Number of times reached maximum'

  Note that RAPOOL only applies to TORs, not for MRO sessions between CICS regions. So check and potentially reduce RAPOOL for FORs/AORs.

SIT SEC ...

SEC

SEC=YES!NO	Security by BSM or ESM

Measurement Example:

Using SEC=YES with a single DSW partition showed ...

STATRCD

STATRCD=ON	Statistics Recording control

Status of the CICS interval statistics.
Does not apply to requested or shutdown statistics.

STGPROT

STGPROT=NO	Storage Protection facility

STGPROT=YES (default is NO) activates the CICS Subsystem Storage Protection, which separates the CICS code from applications.

Make sure, CICS TS does NOT run in F4, when you want to use it.

Measurement Example:

Using STGPROT=YES with a single DSW partition showed no measurable overhead.

- **Use Subsystem Storage Protection with its huge benefit**

  except a vendor program would not allow (nothing known to us).

  Contact Vendor and IBM, in case you cannot benefit from it

SIT STNTR ...

STNTR

STNTR=1	Standard Tracing

Specifies the level of tracing. 1 is default.
OFF disables standard tracing.

This applies to Internal tracing.

SVA, PRVMOD

SVA=NO!YES	Use of SVA resident phases
PRVMOD=name,..	Exceptions in case of SVA=YES (not in DFHSIT)

Specifies whether optional (not SVA mandatory) phases can be used by CICS TS from the SVA.

It applies to

- CICS phases,
- user CICS programs,

AND - any other phases used in the CICS TS partition (e.g. LE, vendor modules, etc)]]

PRVMOD selectively specifies those modules, for which SVA=YES does NOT apply ('Use a non-SVA-copy')

SVA=NO is the SIT default.

Refer to the detailed discussion 'CICS TS SVA Content'

SIT TCTUALOC ...

TCTUALOC

TCTUALOC=BELOW	TCTUA Storage location

Specifies the location of the TCTUAs (terminal control user areas).

By moving the terminal control blocks above the line,
about (600 +TCTUAL) bytes can be saved per installed terminal.

TCTUALOC=BELOW is required for all programs that cannot access areas above the line and which use TCTUA.

- **Leave TCTUALOC=BELOW for all CICS TS partitions that use ICCF and/or IUI.**

  Only use TCTUALOC=ABOVE when you are really sure that

  - no ICCF is included in your CICS
  - no IUI function can be invoked
  - no other program needs it

TD

TD=(3,3)	Transient Data buffers and strings

Specifies the number of TranData buffers and strings for the TranData INTRA file

- **Leave defaults. Use statistics to increase if req'd**

TRT ...

TRTABSZ=80	Internal Trace Table size in K
TRTRANSZ=80	Txn Dump Trace Table size in K
TRTRANTY=TRAN	Txn Dump Trace type

The Internal Trace Table is located above the line, and always a multiple of 4K.
Its minimum (and default) value is 16 (K).
Use at least 5 times the size from CICS/VSE

It is allocated permanently, and only used if Internal Trace is on.

The Txn Dump Table is also located above the line, its default value is 40K.

It is only allocated (VSE GETVISed) during a txn dump to make a copy of TRTAB, which:

- Avoids overlay of trace entries during Dump
- Reduces Dump impact on CICS throughput

TS

TS=(,8,8)	TempStor buffers and strings

Specifies the number of AUX TempStor buffers and strings.
TS=(,0) enforces MAIN TempStor.

- **Leave defaults. Use statistics to increase if req'd**

- **Try TS=(,0) if you do not have large requirements for TempStor (maybe <20M) and no need for recovery**

  See 'Peak Number of CIs in Use' and the CI-size to determine requirements

SIT TSMGSET ...

TSMGSET

TSMGSET=20	TempStor MessaGe SET parameter

Specifies the number of record item slots in a single TSGID control block, used for any TempStor MAIN and AUX queue and built when queues are created and extended.

Allows some trade-off between CPU-cycles and virtual storage.

Refer to the TempStor Statistics, showing the number of queues and the number of extensions created.

- **If more than 10% of the total queues created require extension, increase TSMGSET**

WRKAREA

WRKAREA=512	Commom workarea (CWA) size

Size of the CWA potentially used by applications.
Resides below the line.
If not used, it can be set to 0.

XRF

XRF=NO	Extended Recovery Facility option

Specify XRF=NO if XRF is not used.

Even if no alternate CICS partition is up, a small virtual storage overhead is expected for an XRF=YES specification:

XRF trace table (e.g. 64K, where??)
Addt'l modules DFHWSMS/DFHWSSON (73K 31-bit)

The CPU-time overhead of XRF usually is around 1% to 2%.

This applies to the 'surveillance and tracking stage', where surveyance signals are written to and read from the CAVM (CICS Availability Mgr).

Check the use of VTAM Persistent Sessions in order to reduce CICS TS TOR system outage times, and to offload the network.

PART I.

CICS TS Measurement Results/Systems

General Remarks

Measurement Tools/Methods

For CICS TS performance runs, the following Tools/methods were used

QUERY TD,INTERNAL
QUERY TD,INTERNAL + SIR MON=ON (some overhead)
VSE Job Accounting (JA)

AUXtrace
SDAID
DEBUG SHOW
SIR SMF

DFH0STAT or DFHSTUP

In addition, for selected runs a H/W monitor was used, in order to check TD CPU-times and VSE JA data.

„ All runs were done with QUERY TD,INTERNAL

In limited cases SIR MON was used, either on top, or just for getting relative results as only run.

SIR MON overhead is mostly <10%, so the relative delta vs a 2nd run with also SIR MON was only marginally impacted.

„ Standard Scheme

```
SIR MON=ON                    (if used)
SYSDEF TD,RESETCNT
////////////////////// activity
QUERY TD,INTERNAL
SIR MON                       (display, if used)
```

Sometimes SIR SMF for I/O counting was used on top.

„ Processor Environment

For all runs, a 9672-R61 CMOS processor was used, especially calibrated and equipped with a H/W monitor.
All measurement results shown here were obtained in a VSE/ESA native environment (i.e. without VM/ESA) on either 1-way or 2-way.

General Remarks ...

CICS Total Workload Runs with 'DSW'

- **As in the past, the Data Systems Workload 'DSW' was used**

  All users were simulated via TPNS on a separate processor, each with 11 sec thinktime.

  1 CICS partition, 300 users (on 1-way)
  3 CICS partitions, 3x300 users (on 2-way)

  For SIT parameter variations, 1x300 user were taken, apart from a comparison of 2x300 vs 1x600 user on 1-way.

Software Used

- **Many VSE/ESA 2.4 and CICS TS drivers were used**

- **Performance improvements went into later drivers**

- **Most runs were done (12/98)**
  with the 'LA-driver'
  PLUS 3 performance PTFs,
  not contained in VSE/ESA 2.4 (12/98) driver,

  but being part of one of the 3 PTFs for VSE/ESA 2.4.0 LA:

  PTF1 APAR DY44858, PTF2 DY44889, PTF3 45037 (GA, 06/99).

  They refer here to VSE/ESA:
  - Improved processing of MVS WAITs, issued by CICS TS
  - Improved ESTAEX Program Call
    (used each time an ABEND is to be intercepted)
  - Faster dispatch of CICS TS

  Also, CICS TS performance PTFs will be included:
  - dispatching
  - monitoring
  - program control

- **Also 04/99 runs were done with a GA equiv. driver**

General Benchmark Characteristics

Workload Characteristics

(here: General View for Processor Evaluation, LSPR)

- Relative I/O intensity
- Relative dispatch intensity
- Frequency and spectrum of supervisor calls
- Ratio of supervisor to problem state time
- Instruction mix incl. instructions costly as VM guest
- High speed buffer hit ratio
- Branching behavior
- Storage reference pattern
- N-way exploitation capability

Refer e.g. to Large System Performance Reference (LSPR)

OLTP Benchmark (DSW)

DSW (Data Systems Workload)

- **Transaction Mix**

  20% Banking and teller system

  (0-10 logical I/Os)

  6% Hotel reservation

  (2 logical I/Os)

  17% Inventory tracking

  (0-17 logical I/Os)

  20% Order entry

  (3-22 logical I/Os)

  19% Product specification

  (18-34 logical I/Os)

  18% Stock control

  (3-18 logical I/Os)

- **Per average transaction:**

  3.5 BMS requests
  6.9 VSAM requests
  1.0 Transient Data or Temporary Storage requests
  6.1 Journal requests
  7.2 GETMAIN/FREEMAIN request pairs

- **Pseudo-conversational**

  CICS storage not held during user thinktime

- **Uses a broad spectrum of CICS services**

- **With specific setup also used for VSE LSPR**

OLTP Benchmark (RAMP-C)

RAMP-C

(Requirements Approach to Measure Performance in COBOL)

So far not yet used for CICS TS.

File functions under CICS TS were individually measured with a newly extended CICS transaction (see VSAMTEST).

„ Synthetic interactive commercial application

35% simple

(1 indexed read, 1 sequential add)

25% average

(3 indexed, 2 seq. reads, 1 indexed update, 1 seq. add)

30% complex

(11 indexed, 5 seq. reads, 6 indexed updates, 1 seq. add)

10% very complex

(25 indexed, 5 sequ.reads, 10 indexed updates, 1 seq. add)

	Think-time	Log. R/W ratio
Class 1	8 sec	1:1
Class 2	10 sec	3:1
Class 3	18 sec	3.14:1
Class 4	26 sec	11.1:1

„ Conversational

CICS/VS storage held during user thinktime

„ Very VSAM file intensive

DIM setup reduces I/O intensity, but not logical file intensity

CICS SIT Options for DSW Runs

	CICS/VSE	CICS TS	Exceptions/Variations
Partition Size	50M	50M	
EXEC DFHSIP SIZE	DFHSIP	DFHSIP	
	SUFFIX=23	SUFFIX=41	
Statistics	done always	DFH0STAT	DMF started on top
Monitoring	no, MONITOR=NO	MN=OFF	
Journaling	JCT=NO	JCT=NO	DSW may use 'JCT21'
	AKPFREQ=200	AKPFREQ=0	
MRO/ISC	ISC=NO	ISC=NO	ISC=YES
	IRCSTRT=NO	IRCSTRT=NO	IRCSTRT=YES
	MROBTCH=1	MROBTCH=1	MROBTCH>1,
	MROLRM=YES	MROLRM=YES	MROLRM=NO
LESTG	LESTG=4M	not appl./req'd	
Security	SEC=NO	SEC=NO	SEC=YES
Tracing	TRACE=OFF	INTTR=OFF	INTTR=ON
	AUXTR=OFF	AUXTR=OFF	AUXTR=ON
Intervals	ICV=1000	ICV=1000	ICV=varied
	ICVTSD=200	ICVTSD=200	ICVTSD=0
Areas	n/a	DSALIM=5M	
		EDSALIM=25M	
Txn Backout	DBUFSZE=2000	DBUFSZE=2000	
	DBP=1$	DBP=1$	
Tasking	MXT=999	MXT=80	

Run Matrix for Variations with CICS TS

Results of DSW Runs

In order to do VSE and CICS release regressions, 'standard' runs with DSW were done with

VSE/ESA 2.3	CICS/VSE 2.3
VSE/ESA 2.4	CICS/VSE 2.3
VSE/ESA 2.4	CICS TS 1.1

Refer to: VSE/ESA 2.4 part 'VSE/ESA Basic CPU Requirements'

Run Matrix for Variations with CICS TS

Runs done/planned	Change via	
DSW 'standard' with CICS/VSE	-	OK
DSW 'standard'	-	OK
DSW but Int. Trace on (INTTR=ON)	CEMT	OK
DSW but w/ DMF started	directly	OK
DSW but w/ real Journaling (JCT=xx, FCT=xx w/ recoverable files)	SIT ovwr	
DSW but w/ AKPFREQ=200	SIT ovwr	
DSW but ICV=small/large	CEMT	OK
DSW but with ICVTSD=0	CEMT	OK
DSW but with ISC=YES	SIT ovwr	OK
DSW but with SIT SEC=YES (SYS SEC=NO)	SIT ovwr (SYS SEC=YES)	-
DSW but with STGPROT=YES	SIT ovwr	OK
DSW but with CICS Monitoring ON (MN=ON,MNCONV=YES,MNEXC=ON, MNFREQ=10000,MNPER=ON, MNSYNC=YES)	CEMT	OK

Note:

All DSW runs done here are in stable state (fixed # terminals w/o new LOGONs)

So SIT variation result values shown are examples

CICS File Function Measurements

In order to check the important VSAM file functions, we used our CICS transaction VSAMTEST in both CICS releases, and with different setups.

VSAMTEST

This transaction uses the program VSAMTEST, and the mapset VSAMSET. It provides a number of file functions as identified on the displayed screen:

R READ the specified record 10 000 times.
When run with SHAREOPTION(2) this normally does no file I/O after the first execution, as the data will be in a Data Table or in VSAM buffers. With SHAREOPTION(4), there will be 20 000 physical READ operations (potentially hardware cached).

U READ and ReWRITE the specified record 10 000 times.
When run with SHAREOPTION(2) this normally does 10 000 physical WRITE I/Os. With SHAREOPTION(4) there will be 30 000 -- 20 000 physical READs and 10 000 physical WRITEs.

(A) Add a new record (record content in record area on screen).

(D) Delete a record.

B (StartBrowse, ReadNext, Endbrowse) 10 000 times
With SHAREOPTION(2) this will do no physical I/O.
With SHAREOPTION(4) this will do 20 000 physical I/Os.

L (StartBrowse, ReadNext 10 times, Endbrowse) 1 000 times.
With SHAREOPTION(2) this will do no physical I/O.
With SHAREOPTION(4) this will do 2 000 physical I/Os.

Normally, only R, U, B, and L functions are used in testing.

Use the Clear Key to exit. This program is 'pseudo-conversational' and has no special 'memory' effects, so you may exit the process at any time.

Í VSAMTEST results simply show the CPU-time for the CICS function, no application logic included.

Overview on VSAM File Function Results

		VSAM LSR (local) Rel_CPU	IO	MRO FS (remote) Rel_CPU	IO	VSAM LSR Data (local) Rel_CPU	IO	Table Data Table (remote) Rel_CPU	IO	
CICS TS SHR2 U	R	1.00	0	28.75	0	0.50	0	0.72	0	
	<7.54	1	<66.05	1	<8.20	1	<66.77	1		
	B	3.44	0	86.11	0	1.26	0	1.66	0	
	L	0.87	0	33.93	0	0.60	0	0.84	0	
CICS TS SHR4 U	R	<11.38	2	<39.13	2	0.51	0	0.72	0	
	<19.37	3	<77.89	3	<20.25	3	<78.81	3		
	B	<14.02	2	<96.69	2	1.26	0	1.66	0	
	L	< 1.92	2/n	<34.99	2/n	0.60	0	0.60	0	
CICS/VSE SHR2 U	R	1.32	0				0.22	0	MRO FS	
	7.10	1	later		7.53	1	also for			
	B	3.79	0				3.78	0	READs	
	L	1.02	0				1.02	0	later	
CICS/VSE SHR4 U	R	10.28	2			0	0.22	0		
	17.37	3	later		18.45	3	- " -			
	B	12.90	2			(13.04	2)			
	L	1.94	2/n			(1.95	2/n)	later		

- local: access to data owned by the same CICS partition
- remote: access to data being owned by another CICS partition

- All Data Tables were CICS Maintained (CMTs)

- Data Table 'remote' Updates include Function Shipping

- No MRO mixture of CICS TS and VSE here

- Functions measured ('RUBL') are
  READ/READUPDATE/BROWSE/LONGBROWSE

- Rel_CPU values are resulting total RELATIVE CPU-times
  (for illustration, here 10000 functions were used)

- IO values represent the measured number of I/Os per function
  (x/n for LONGBROWSE means: n GETNEXTs between START and ENDBROWSE).
  Number of I/Os clearly illustrate duration of function

- All CICS TS functions with I/Os represent worst case figures.
  In a CICS partition with other transactions active, less CICS MVS WAITs would be given

CICS File Function Measurements ...

VSAM File Function Results (CICS TS vs CICS/VSE)

Preliminary RELATIVE CPU-times, will be complemented/updated

CICS TS vs CICS/VSE		VSAM LSR (local) MRO Rel_CPU	VSAM LSR Data FS (local) (remote) Rel_CPU	Table Data Table (remote) Rel_CPU	
SHR2	R	0.76 *1		2.27 *3	
	U	<1.06 *2 later	<1.09 *2	later	
	B	0.91		0.33 *4	
	L	0.85		0.59 *4	
SHR4	R	<1.10 *2		2.32 *3	
	U	<1.11 *2 later	<1.10 *2	later	
	B	<1.09 *2		0.10 *4	
	L	<0.99		0.31 *4	

- Same explanations as for table on last foil
- Always keep in mind the importance (frequency) of a function and setup

- **The most important value (SHR2 READ) is much better in CICS TS (*1)**

- **Most functions which include CICS WAITs have increased, here (*2)**

  In a CICS partition with other transactions active, these ratios would be better, since CICS MVS WAITs would be given less frequent (< signs).

- **READ Data Table access has increased (expected), but still is much better than VSAM LSR READ (*3)**

  Similar degradation as was seen in MVS.
  Data Tables in practice are even more beneficial, since here only a very small VSAM LSR subpool was used

- **Any BROWSE of a Data Table is better than the BROWSE in CICS/VSE (*4)**

CICS File Function Measurements ...

VSAM File Function Results (SDT vs VSAM LSR)

Shared Data Tables vs VSAM LSR (Local and Remote)

Preliminary RELATIVE CPU-times, will be complemented/updated

CICS TS SDT vs VSAM LSR		Local Rel_CPU	Remote Rel_CPU
SHR2	R	0.51	0.03 *5
	U	1.09	1.01
	B	0.36	0.02 *5
	L	0.70	0.03 *5
SHR4	R	>0.05	>0.02 *5
	U	1.05	1.01
	B	>0.09	>0.02 *5
	L	>0.31	>0.02 *5
- Same explanations as for last tables - Always keep in mind the importance (frequency) of a function and setup			

- **Any Data Table READ is much better than VSAM READ**

  This, naturally, includes BROWSE, now.

- **For WRITEs (READUPDATE) slightly more CPU-time is required**

  Data Table must be updated, on top of VSAM LSR access.

- **HUGE benefits for cases where no Function Ship is needed anymore (*5)**

CICS TS DSW Workload Variation

Vary number of users and CICS TS partitions

Part. x User	CPU util	NPS	txn/sec	Av. RT (sec)	CPU-time /txn(ms)	ITR-ratio		
1 x 300	36%	0.290	20.33	0.073	17.92	1.00		
2 x 300	70%	0.274	39.88	0.117	17.93	1.005		
1 x 600	69%	0.263	40.86	0.117	17.16	1.045		

- Standard system setup and environment
- IO/txn very constant at 4.3
- No MRO was used here

Measurement Conclusions:

As expected ...

CICS TS (also CICS/VSE)

- has no problems for single partitions

(as long as partition capacity is big enough)

- has no problems for multiple partitions

Only a small overhead in CPU-time if a total workload is split into several CICS partitions (w/o need for MRO)

- **Run as much as you want/can in 1 CICS partition**

  (CICS TS partition capacity is higher than for CICS/VSE, due to VSCR).

On n-ways, multiple CICS partitions are of benefit

OPEN

„ Cost (in terms of CPU-time) of transferring a record to DMF should be less than the cost of processing a CICS journal record

„ CICS Monitoring previously used Journaling.

Does that mean that by using DMF a lower overhead should show up for monitoring ?? tbd

„ Will MRO performance change ?? tbd

„ What FUNCTIONAL reasons does the ICV interval still have in CICS TS??

What would happen If I select the largest ICV value possible?? (MROBTCH impact)
- CSFU txns
- CSNL console txns
- terminals having no ECB ??
- CICS shutdown

„ Faster CICS startup

More activities done in (I/O) overlap YES, by design, by more real subtasks
More warm start data provided and is less data is to be read from CICS tables.

„ Faster CICS shutdown

Are RSD updates done earlier??

„ Can Int. Trace overhead be reduced/avoided ?

(can ISC/MRO system related functions be disabled for internal trace ?) YES, see CETR ISC (also valid for MRO ??)

„ Do CICS Stats contain

- Info about individual file/database accesses? YES
- Info on which txn needs which file, for MRO tuning NO (only contained in Monitoring results)
- LSR buffer hits YES
- LSR buffer hit ratios, NO
  # log. requests and physical I/Os per file YES

PART J.

App. M: VSE/ESA 2.4 Perf. Enhancements

Note:
For practical reasons ...
this part currently is hosted in this document

VSE/ESA 2.4 Perf. Items (Summary)

VSE/ESA 2.4.1 refresh is available since 12/99

- **All Performance PTFs shipped after V2.3 GA**
- **VSE Base related**

  ESA/390 only

  ESA/370 not sufficient.

  Basic changes for CICS Transaction Server (TS)

  'OS/390 Emulation Mode' and more (see below)

  Turbo Dispatcher Only

  Setup Items
  - **Label Area on 'Virtual Disk' now standard**
  - **DASDFP now standard**
  - **250 MB VSIZE shipped layout**

    Was 180 MB, Environment B

  Fast CCW Translation dropped

  Improved Display of SVA content

- **VSE/POWER related**

  POWER Queue File above the line
  Higher SETPFIX limit
  Up to 32 POWER Data File extents

VSE/ESA 2.4 Perf. Items (Summary, cont'd)

- **VSE/VSAM related**

  Support of separate Index and Data LSR (sub)pools

  Use of Track Caching for all ESDSs

  (by default also for DL/I and DB2/VSE)

- **CICS TS for VSE/ESA related**

  CICS/ESA for MVS 4.1 functionality and more

  Family API / OS/390 Emulation Mode

  Support of more ESA/390 functions:

  - **Space Switching Program Call**
  - **Cross Memory Mode**

    Communicate between partitions w/o system services
  - **Subsystem Storage Protect**

    SSP is only required if used.
    Available on all ES/9672 processors,
    and all ESA/390 capable ES/9221 models.
    On 9121 processors after 92-06-26,
    before that date only via RPQ.

- **Misc. Product related**

  New DL/I 1.11 release

  IXFP/SnapShot for VSE/ESA, to exploit RAMAC Virtual Array

  BTAM/ES 1.1.0 as optional product

  CICS/VSE 2.3 only

VSE/ESA 2.4 Base Enhancements/Items

„ Turbo Dispatcher Only

Now TD functions are always available, including those for unis:

- **Same Part. Bal. weight for Dynamic and Static partitions**
- **Relative VSE SHAREs**
- **QUERY TD Display**
- **SVC Monitoring**

Still, do NOT define more processors than available/needed.

Be aware of additional CPU-time required for the TD, if so far only the Standard Dispatcher (SD) was used.

„ DASD File Protect now standard

Is now standard in the shipped IPL procedure. No change of the default.

- **In order to use DASDFP (recommended), leave DASDFP=YES in the ASI procedures**

Measured overhead in an I/O intensive Batch workload (PACEX) was very small (+1% CPU-time only).

„ Label Area on VSE 'Virtual Disk' now standard

This feature had to be activated formerly by customer setup.

It was very simple for the customer to activate it after IPL, so we decided only now to make this the standard.

Now address FDF is used.

Keep in mind, that in VSE/ESA V2, the label area is accessed directly in the data space, without using CCWs for a virtual disk.

VSE/ESA 2.4 Base Enhancements/Items (cont'd)

„ Fast CCW Translation dropped

In order to streamline and simplify VSE/ESA I/O handling, the VSE Fast Translate option was dropped.

The only cases, where Fast CCW Translation was beneficial were cases, where, repetitively and in a fast sequence,

- the very same channel programs, at very same location, with the very same data buffers

were used.

This mostly applies/applied to BAM, only seldom to VSAM.

Examples

Former Benefits/Costs of FASTTR	
Type of job	CPU-time delta
COBOL Compile and Link	- 3% (saved)
VSAM production job	-2% to +5%

For all other access methods, specifying FASTTR, usually represented a CPU-time overhead:

If no replicas were found, a long scan was involved, until 'normal CCW translation' was done anyhow.

As a consequence, BUFSIZE related values in the SIR display can be better interpreted and adjusted:

COPY-BLKS = 00459 HIGH-MARK = 01243 MAX = 01500

More Information

For description of the now obsolete FASTTR function, refer to
'IBM VSE/ESA 1.3/1.4 Performance Considerations'
Foil: To 'Fast' or 'No-Fast Translate' ?

VSE/ESA 2.4 Base Enhancements/Items (cont'd)

- **Improved Display of SVA content**
  - **LD SDL Display**
    - **- SVA/MOVE MODE changed from YES to 24/ANY/MOVE**

      This shows for the first time the eligibility of an SVA phase regarding the location within the SVA
    - **- LOADED INTO SVA changed from YES to 24/31**

      No need anymore to interpret the address in the SVA/VLA to see whether it resides above or below the line
    - **Easily detect 31-bit eligible phases wasting 24-bit space**
  - **New info msg, if a phase RE-loaded into SVA**

    L168I PHASE phasename RELOADED INTO SVA-xx - PREVIOUS LOC. SVA-yy
    - **Easily detect multiple loadings of SVA phases, so far not visible from LD SDL**

      Multiple loading may occur (and is required) during service application, but NOT on a permanent base

VSE/ESA 2.4 Base Enhancements/Items (cont'd)

„ **New DUMP Options in STDOPT and OPTION statement**

Prevent Dumps from being written to SYSLST, if dump library is not available:

(full / in error / not defined).

STDOPT SYSDUMPC=YES (system wide) OPTION SYSDUMPC (job JCL)
- Both options must be set to avoid that dumps are spooled to SYSLST - OPTION NOSYSDUMP overrides STDOPT SYSDUMPC=YES for a job step

í **Use these options to avoid unexpected blocking of a whole partition (e.g. CICS)**

CICS TS Related Enhancements

CICS TS Related Enhancements/Items

- **CICS Transaction Server for VSE/ESA 1.1**

  Functional level as of CICS/ESA for MVS 4.1 + PTFs

  - **Refer to the new document**

    'VSE/ESA CICS Transaction Server Performance'

- **'Family API' for 'OS/390 Emulation Mode' includes permanent use of 'Cross-Memory mode'**

  - **These functions are only available in ESA/390**

    ESA/370 processors no more supported
    (e.g. 4381-9XE, ES/3090...)

More info on next foils

Background Info

- **The common code base for CICS/ESA in MVS (OS/390) and VSE is possible, since ...**
  - **VSE/ESA 2.4 accepts certain MVS SVCs**
    - **Some are interpreted**
      (and translated as into VSE SVCs)
    - **Some are implemented natively**
    - **Only a subset of MVS SVCs can be understood by VSE/ESA 2.4**
      Only available for CICS TS, DMF and specific vendor products
  - **VSE/ESA 2.4 supports the space-switching Program Call (PC-ss)**
    Refer to the separate foil
  - **VSE/ESA 2.4 also exploits Subsystem Storage Protect (SSP)**
    Required if CICS TS uses SSP (SIT STGPROT=YES), to separate CICS code from user applications.
- **Any job requiring OS/390 emulation mode must run with EXEC phasename,...,OS390**

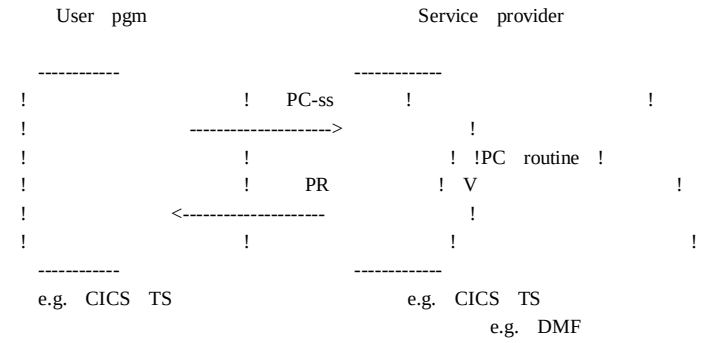
Causes to setup a pseudo-MVS environment:

- OS/390 type of control blocks (incl. TCBs) in SVA-24 and -31
- Allows use of MVS-like macros and PC-services (Family API, Cross-Memory services)

You should not specify it, except when required.

Space-Switching Program Call

Cross memory communication enables a program to transfer control to another program in another address space:



The PC routine executes under the same task as the user program.

Used e.g. by

- CICS Data Management Facility (DMF) server

  (CICS TS calls a DMF routine)

- CICS Shared Data Table (SDT) server

Naturally, VSE Job Accounting (JA) correctly associates all CPU-time to the originating partition/task.

VSE/ESA 2.4 VSAM Enhancements/Items

„ Separate LSR (sub)pools for Data and Index

In total, up to 15 LSR pool-IDs are possible, but each pool may consist of a pool for Data and a pool for Index.

Each pool consists of subpools; a subpool consists of all the buffers for a specific CI-size.

Í 2x15 pools effectively

í More DIM possible, and/or shorter subpools

In order to exploit separate (sub)pools for Index and Data, the BLDVRP macro has been extended:

BLDVRP ...,TYPE=LSR	Creates a single common pool (as used previously)
BLDVRP ...,TYPE=LSR,DATA	Creates a pool for data only
BLDVRP ...,TYPE=LSR,INDEX	Creates a pool for index only

CICS TS exploits the extended BLDVRP macro, but also any other program (e.g. Batch) could exploit it.

„ VSE/VSAM Track instead of Record Caching (RC)

for ESDSs OPENed with CNV and UBF (includes DL/I and DB2/VSE)

For all VSAM ESDS files on ECKD attached devices using Control Interval Access (CNV) and User Buffering (UBF), the default of using RC was changed.

With the bigger DASD caches, using RC started to turn out to become less effective for some I/O subsystems.

í By default, Record Caching is no more used

Via a new bit in the COMREG (IJBRECCA), VSAM still can use RC (in a specific customer environment).

For more info on that, refer to
'DASD Cache Bit Settings for VSE/VSAM' in
'IBM VSE/ESA I/O Subsystem Performance Considerations'

APAR DY44796 (PTF UD50775) retrofits this to VSE/ESA 2.x

VSE/POWER Enhancements/Items

„ POWER Queue File Virtual Copy above the line

This function was of urgent need for customers needing or using a huge POWER Queue file,

- by normal growth
- by using SYSLST queue as a repository
  (keeping job listings, if requested later again)

Refer to 'POWER GETVIS Requirements' chart in the VSE/POWER part of the VSE/ESA V2 Base document.

„ Maximum POWER SETPFIX LIMIT of 2M

The former maximum of 1M sometimes was too small

- when too many 'real' (=PFIXed) I/O buffers were required

and/or

- when for DBLK a bigger size was selected (e.g. track/2)

í Monitor 'real' requirements via MAP or PDISPLAY

Avoid a too high SETPFIX limit

- does NOT cost directly real storage
- BUT: costs virtual storage in VSE/POWER partition
  (in contrast to normal usage)

„ Up to 32 POWER Data File extents

So far, only 15 POWER Data file extents were possible.
This enhancement is available only via PTF (APAR DY45190) on top of VSE/ESA 2.4.0 GA level as of 99-06-25.

The number of POWER Data File extents used is now shown in the D STATUS display.

Misc. Product Enhancements

„ DL/I VSE 1.11 (New Release)

Required for DL/I environments with CICS TS.
Recommended for pure DL/I Batch environments.

DL/I 1.11 exclusive functions are described in

'IBM VSE/ESA CICS Transaction Server Performance'

The effect of the performance PTFs also available for DL/I 1.10 are described in

'IBM VSE/ESA V2 Performance Considerations'

„ IXFP/SnapShot for VSE/ESA

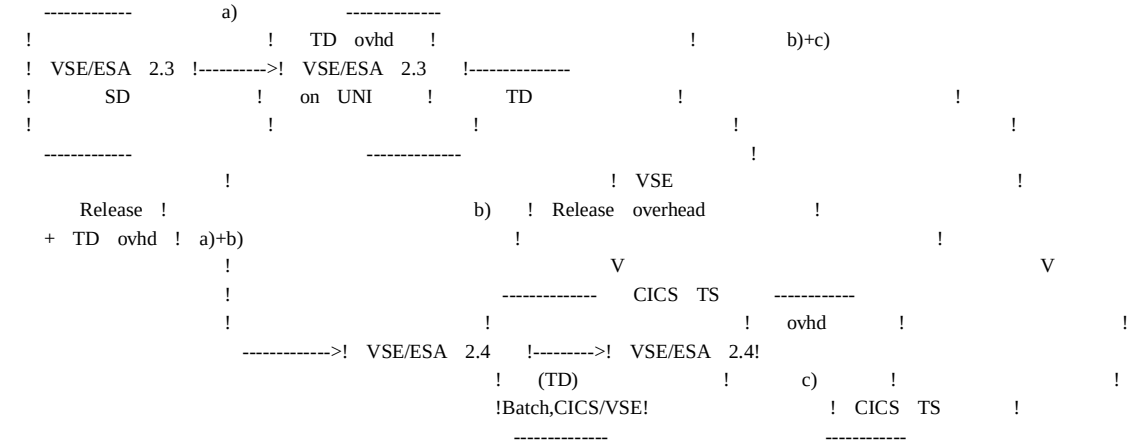
Allows to 'instantly duplicate data' on RAMAC Virtual Array (RVA) I/O subsystems.

Is also available in VSE/ESA 2.3, via separate PTF.

Documented e.g. in

'IBM VSE/ESA I/O Subsystem Performance Considerations'

VSE/ESA 2.4 Basic CPU Requirements



a) TD Overhead on UNI

These deltas refer to VSE/ESA 2.1, 2.2 or 2.3.

About 10% CPU-time overall,
depends on properties of a total workload.

Measurement results

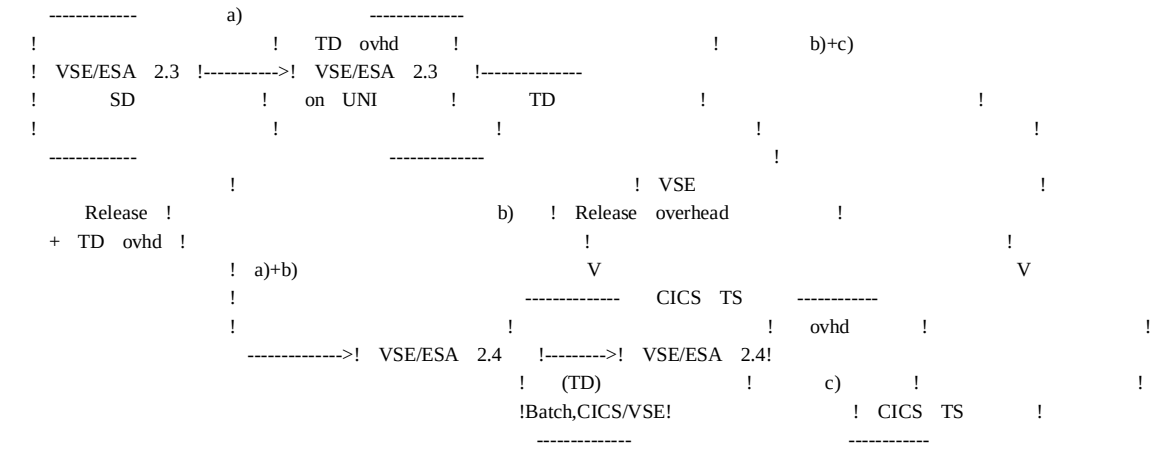
PACEX Batch (I/O and supervisor intens.) up to +15%	
DSW CICS/VSE Online (CICS function intens.) + 4% RAMPC CICS Online (VSAM intensive) + 7%	

For individual, non-representative jobs, higher individual values may be observed.

Refer to the extensive discussion in

'IBM VSE/ESA Turbo Dispatcher Performance'

VSE/ESA 2.4 Basic CPU Requirements ...



b) VSE Release Overhead

The following CPU-time deltas are expected between VSE/ESA 2.4 and VSE/ESA 2.3 (both run with Turbo Dispatcher):

About 5% for 'average' overall workloads, depends on properties of a total workload.

'Average' means:

- a total load, which is not too I/O intensive
  (i.e. where some Data In Memory has been already applied)
- excluding I/O intensive single jobs and transactions

Measurement results

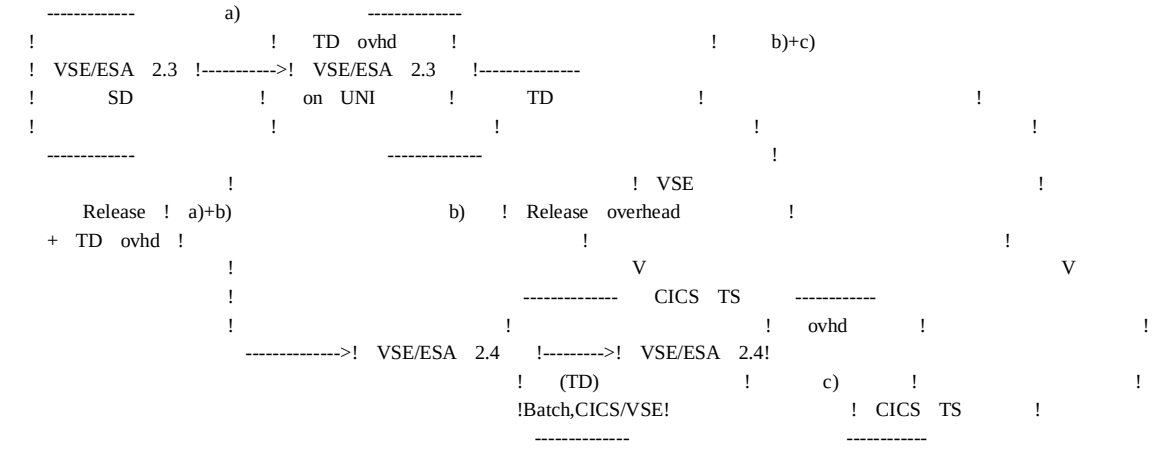
PACEX batch (I/O and supervisor intens.)	+4% (1 part.)	+6% (16 part.)
DSW CICS/VSE Online (CICS function int.)	+7% (1 part.)	+5% (3 part.)

For individual, non-representative jobs higher individual values may be observed.

Reasons for longer pathlength:

- More functions in VSE/ESA 2.4
- Permanent use of Cross Memory mode required
  (also Control Registers more actively used)

VSE/ESA 2.4 Basic CPU Requirements ...



c) CICS TS Overhead

About 5% CPU-time overall,
depends on properties of a total CICS workload
(w/o exploiting new perf functions).

Measurement results

The following CPU-time deltas were observed, regarding the use of CICS TS vs CICS/VSE (both under VSE/ESA 2.4)

DSW CICS Online (CICS function intens.) +6% (1 part.)	+4% (3 part.)

For individual, non-representative txns higher values may be apply.

Reasons for partially longer pathlength:

- Much more CICS TS functions
- More control blocks in OS/390-mode
- Interpretation of MVS SVCs
  (especially MVS WAIT/POST, ATTACH/DETACH)

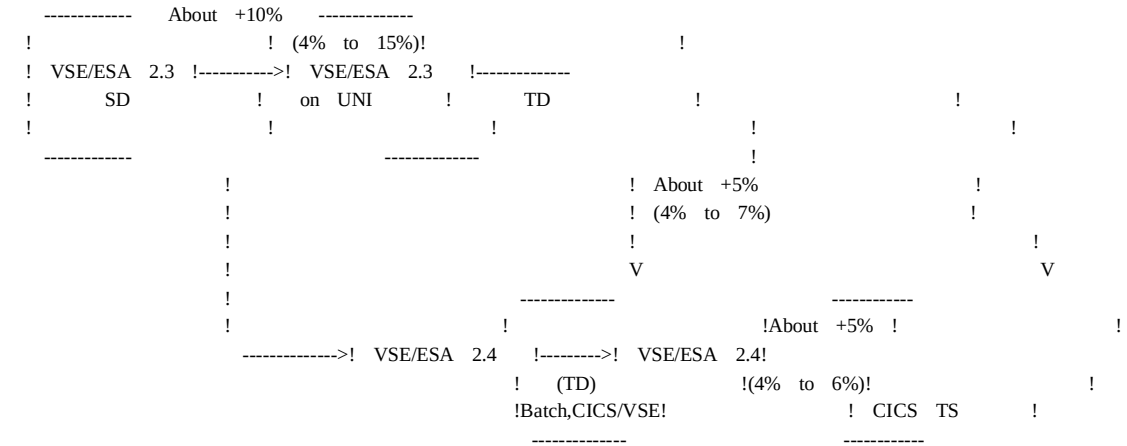
Many reasons exist for partially SHORTER pathlength, reducing overall CICS TS overhead (new 'performance functions').

Refer to the extensive discussion in
'IBM VSE/ESA CICS Transaction Server Performance'

VSE/ESA 2.4 Migration (SUMMARY)

CPU-time Requirement Deltas (STATUS 04/99)

Average I/O intensive, misc. total workloads:



Total percentages for individual workloads;

Workload	With TD 'migr.'	w/o TD migr.	
PACEX (I/O intens. Batch) 15% + 5%	= 20%	5%	
DSW (CICS fct. intens.)			
CICS/VSE 4%	+ 6%	= 10%	6%
CICS TS 4%	+ 6% + 5%	= 15%	11%

- **Figures hold for 'average' workloads**

  Very I/O intensive (non-'DIMed') workloads/jobs may show higher values

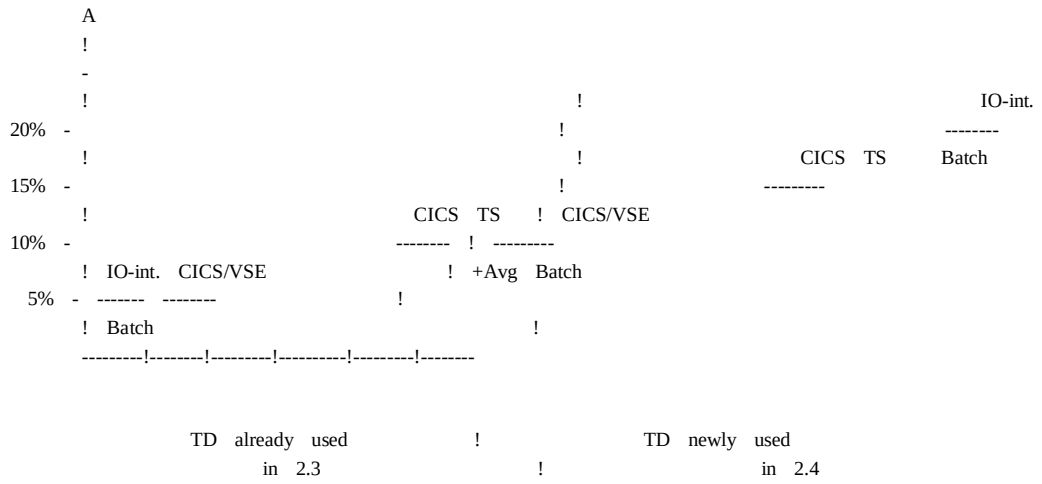
- **Add up percentages for YOUR migration**

- **Make sure, before Migration, that enough CPU-power is/will be available**

VSE/ESA 2.4 Migration (SUMMARY, cont'd)

Total Percentages for Workloads (RE-VISITED)

Rough CPU-time deltas vs VSE/ESA 2.3:



Refer to previous charts for explanations and caveats

Migration to VSE/ESA 2.4

- **As for any other VSE Release Migration....**
  - **Produce and keep 'old' Monitoring/Performance data**
  - **Carefully evaluate CPU requirements**
    - **When skipping several VSE releases**
    - **When you are CPU-bound in the peak hour**
    - **When you do not want/can exploit Data In Memory**
  - **Distinguish between**
    - **- 'first day performance' and a tuned system**
    - **- individual job figures and overall figures**

If you migrate from VSE/ESA V2 to VSE/ESA 2.4 ...

- **Use the TD first on your 'old' release before you migrate**
  - **Migration to CICS TS should start from CICS/VSE 2.3**
- **More Information**

For general aspects regarding VSE migration, refer to

'IBM VSE/ESA Hints for Performance Activities'.

Part E: Pure Processor Migration Aspects
Part F: VSE Sizing Checklists
Part G: LSPR Results for VSE/ESA
Part H: VSE/ESA Quick Migration Sizer

EOD HAND

EOD	End of Document
HAND	Have A Nice Day