IBM VSE/ESA 1.3/1.4 Performance Considerations

Wolfgang Kraemer

VSE Product Mgmnt
Dept 3221
71032-04 Boeblingen
WKRAEMER at BOEVM4
wkraemer at de.ibm.com

97-04-29 (no future updates)

Contents

Contents

Contents

What's new?

What has been added/changed?

Deltas as compared to earlier versions

Editorial changes are done throughout the document, without special notice.

Changes for the 10/18/95 update

- Very minor and few changes only

Changes for the 02/26/96 update

- Editorial changes/refinements throughout
- Some new Caching charts
- DOS Assembler problem for VSE/ESA 1.4
- Chart on VM/VSE guest and DASD types
- Inclusion of newer performance APARs and PTFs
- SAP R/2 moved out (to the 2.1 document)

Changes for the 05/03/96 update

- All I/O related charts were moved into a separate document

Changes for the 11/15/96 update

- All VM/VSE related charts were moved into a separate document

No major updates performed in this document since 05/96

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 V1 document

- to appropriately use the performance data.

Unless indicated otherwise, all considerations apply to MODE=ESA supervisors, be it in basic native mode, under VM, or in a PR/SM LPAR.

This document here is unclassified and therefore suited for VSE customers. It is part of the package VE13PERF which resides on an IBM disk, called IBMVSE.

Access to the VE13PERF package is for any IBM person without additional authorization, just by typing

TOOLS SENDTO BOEVM3 VMTOOLS IBMVSE GET VE13PERF PACKAGE (under CMS)

Please contact your IBM representative to obtain an update of this document, if yours is say elder than 6 months.

This document is also available from INTERNET via the VSE/ESA home page

or directly via FTP links

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

Follow-on Documents

'IBM VSE/ESA I/O Subsystem Performance Considerations'
'IBM VSE/ESA V2 Performance Considerations'
'IBM VSE/ESA Turbo Dispatcher Performance'
'IBM VSE/ESA VM Guest Performance Considerations'
'IBM VSE/ESA Hints for Performance Activities'
'IBM VSE/ESA TCP/IP Performance Consideartions'
'IBM DFSORT/VSE Performance Consideartions'
'IBM VSE/ESA CICS Transaction Server Performance'

They are available in VE21PERF PACKAGE on IBMVSE tools disk, and also on INTERNET in Adobe Reader format (.PDF):

VE13PERF.PDF, VE21PERF.PDF, VE21TDP.PDF, VEIOPERF.PDF, VEVMPERF.PDF, VEPERACT.PDF, VETCPPER.PDF, VESORTP.PDF, VECICSTS.PDF

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 better understand IBM products.

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

The above disclaimer is required since for reasons of conciseness not all dependencies can be described in such a type of document.

Acknowledgements

Thanks to all who contributed directly or indirectly, be it by measurements, suggestions or other type of info.

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

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

- suggestions or questions regarding this document
- questions on VSE/ESA performance, not covered in any of the packages (VE12PERF or VE13PERF or VE21PERF PACKAGE)

Note that some additional items are documented in IBM INTERNAL USE ONLY appendages, available to your IBM representative.

Wolfgang Kraemer, IBM VSE Development, Boeblingen Lab, Germany

Notes ...

Base Document

This document essentially deals with the performance differences between VSE/ESA 1.3 and VSE/ESA 1.1/1.2. For VSE/ESA 1.1/1.2 performance versus VSE/SP, refer to the following brochure

'VSE/ESA 1.1/1.2 Performance Considerations'

That document is also available to any IBM person, as part of the VE12PERF PACKAGE on the same IBMVSE tools disk. Contact your IBM SE and ask him to enter the following CMS/VM command:

TOOLS SENDTO BOEVM3 VMTOOLS IBMVSE GET VE12PERF PACKAGE

The latest update of that package is dated 03/31/93.

Subject document contains references to further VSE performance documents.

Trademarks

General Remarks

General Remarks

- This document is technically oriented
- Skip those areas you want, go back if required
- Also aspects are covered which only in certain cases must be understood
- Many tuning knobs are available, but many of them have to be used only
  - if specific problems arise
  - if you need/want to squeeze/exploit VSE and/or your installed H/W to the max
- Some aspects are internally handled by the supervisor, the subsystems and the access methods, but knowledge may help
- We not only tell you where we are better, but also let you understand why

References

Further References

The following are references for further performance information in the context of VSE/ESA 1.3. Also an MVS publication is included, which can be used as examples for Data in Memory (DIM) exploitation to a certain extent.

VSE/ESA 1.3 Presentation Guide
VSEG PACKAGE on MKTTOOLS disk

CICS/VSE 2.2 Performance Guide, SC33-0703

IBM ESA/370, Evolution, Facilities, and Exploitation
ITSC Poughkeepsie, GG24-3303-0, 12/88, 210 pages

MVS/ESA Data in Memory Concepts and Facilities
ITSC Poughkeepsie, GG24-3404-00, 07/89, 27 pages

MVS/ESA and Data in Memory -Performance Studies-
ITSC Poughkeepsie, GG24-3698, 01/92, 336 pages

VM/ESA, Running Guest Operating Systems
Release 1.2.0, SC24-5522-02
Release 2.1.0, SC24-5755-00

VM/ESA, Performance
Release 1.2.1, SC24-5642-01
Release 2.1.0, SC24-5782-00

VM/ESA, Planning and Administration
Release 1.2.1, SC24-5521-03
Release 2.1.0, SC24-5750-03

Maximizing VSE Paging Performance, Enterprise Systems Journal,01/93
pp 52-57, by Justin McMurry

GETVIS in VSE/ESA 1.3, Enterprise Systems Journal,08/94
pp 70-78, by K.-H. Holder and M. Zimmermann

SQL/DS Performance Tuning Handbook for IBM VM Systems and VSE
Version 3, Release 4, SH09-8111-00

VSE/ESA Performance Management and Fine Tuning, by Bill Merrow
McGraw-Hill, 1993, ISBN 0-07-041753-9 or G246-0011-00

SQL/DS Version 3.4 Performance Guide
ITSC Boeblingen, GG24-4047-01 12/94 update with System Implementation chapter, 503 pages

References ...

CICS/VSE 2.2 Data in Memory and Virtual Storage Usage
ITSC Boeblingen, GG24-4185, 12/93, 129 pages

VSE/ESA 1.3, Using the 31-bit Addressing Facility
ITSC Boeblingen, GG24-4191, 12/93, 132 pages

VM/VSE Performance Hints & Tips
ITSC Boeblingen, GG24-4260, 03/94, 110 pages

VSE/ESA 1.3 Virtual Disk as an Additional Tuning Opportunity
VSE/ESA Newsletter 3rd Quarter 1993, G222-4508-07
pp 29-37, by Ulrich Kettner

Exploiting VSE/ESA 1.3: 40% More RAMP-C Throughput
Article by Ulrich Kettner, D/3240 Boeblingen.

As RAMPC PACKAGE on the IBMVSE tools disk,
available for your IBM representative via
TOOLS SENDTO BOEVM3 VMTOOLS IBMVSE GET RAMPC PACKAGE

IBM's VSE/ESA Software Newsletter, 3rd/4rth Quarter 1994, pp 14 .. 23

ES/9000 9221 211-based Models Product Guide
PG21 PACKAGE on MKTTOOLS disk

VM/ESA R2.2 Performance Report
VM Performance Group Endicott, 224 pages.

As VM22PERF PACKAGE on MKTTOOLS disk, latest update 07/15/94,
equivalent to GC24-5673-01

VM/ESA 2.1.0 Performance Report, GC24-5801,
VM Performance Group Endicott, 159 pages.

As VM210PRF PACKAGE on MKTTOOLS disk, latest update 09/15/95,
equivalent to GC24-5673-01

VS COBOL II Rel. 3.2 and 4.0 Performance Tuning 05/94, 21 pages
As COBOLPRF PACKAGE on MKTTOOLS disk,
available thru your IBM representative

Evaluating EXPLORE/VSE in a VM/VSE Environment
ITSO Boeblingen, GG24-4261, 09/94, 89 pages
('Most EXPLORE/VSE values were usable also under VM',
all EXPLORE/CICS values anyhow)

Exploiting VSE/ESA's Virtual Storage Capabilities,
by Dan Janda, VM/VSE Tech Conf Kansas City 05/97, session 33D
and Mainz 06/97, session 53D

Glossary

CICS/VSE 2.2 Performance
VSE/ESA Technical Conference Philadelphia, 09/94

As VSETCPER PACKAGE on the IBMVSE tools disk,
available thru your IBM representative via
TOOLS SENDTO BOEVM3 VMTOOLS IBMVSE GET VSETCPER PACKAGE

VSE/ESA 1.3/1.4 Performance Considerations
by Dan Janda and Mike Augustine, IBM Endicott, WAVV presentation, 10/96, 62 pages
(based mainly on even more extensive document by W.Kraemer)

Glossary

DFW	DASD Fast Write A 3990-3/6 extended caching function
DIM	Data in Memory A concept to store as much data as possible/reasonable in processor storage
EMIF	ESCON Multiple Image Facility Sharing of ESCON channels between PR/SM LPARs
ITR	Internal Throughput Rate A measure for processor and/or S/W effectivity: #transactions or batch jobs per CPU-second
LSR	VSAM Local Shared Resources A VSAM buffering method which allows that different files share the same buffers (Data, Index)
MPG	Multiple Preferred Guest A function on ES/9000 processors, providing improved VM/ESA V=R/F guest support via PR/SM
MRO	CICS Multiple Region Option Provides the required communication of CICS partitions using Transaction Routing (TR) or Function Shipping (FS)
NSR	VSAM Nonshared Resources A VSAM buffering method with separate buffers per file
PR/SM	Processor Resource Systems Manager An ES/9000 standard feature for logical partitioning
VSCR	Virtual Storage Constraint Relief All that provides effectively more space below the 16 MB line

VSE/ESA 1.3/1.4 Performance Overview

PART A.

VSE/ESA 1.3/1.4 Performance Overview

Overview

Ù Capacity Evolution
Ù H/W Support
Ù ESA Exploitation Basics
Ù Overall Performance
Ù Performance Potential
Ù CICS/VSE Capacities
Ù Performance Deliverables and Effect

VSE/ESA Performance/Capacity Evolution

VSE/ESA Performance/Capacity Evolution

```
                                       Even more Real Storage
                                         "    "  Total Virt.Storage
                                         "    "  Private Space
                                       + 31-bit Applications
                                       + 31-bit Buffer Areas
                                       + 31-bit Internal Functions
                                       + Data Spaces
                                       + Virtual Disk
                                       + CICS Data Tables
                                       + Extended Caching Functions
                                    ______________________________________
                                   |
                                   |           VSE/ESA 1.3, 1.4
       More Real Storage           |
         "  Address Spaces         |      -> ESA Exploitation,
         "  Total Virt. Storage    |          mainly for
         "  Partitions             |            - VSCR
         "  Private Space          |            - DIM
        + Dynamic Channel Subsyst. |
        + ESCON                    |
        + 3390 and 9345 DASDs      |
         __________________________|
        |
        |    VSE/ESA 1.1, 1.2
        |
 _______|

  VSE/SP
```


VSE/ESA 1.3/1.4 vs VSE/SP H/W Support

Performance Relevant Processor and I/O Support

„ **ES/9000 uni-processors in basic ESA-mode**

„ **Dynamic Channel Subsystem**

„ **ESCON**
Channels, controllers, devices, directors, EMIF

„ **3990-3/6 Extended Caching Functions**
(including DASD Fast Write, Cache Fast Write)

„ **3390 DASDs (incl. caching)**

„ **9345 DASDs (incl. 'S/W-transparent' caching)**

„ **Up to 1024 VSE local devices**

ESA Exploitation Basics

„ **Do NOT run VSE/ESA 1.3/1.4 with the same setup and parameters as you did before**
(except for temporary migration reasons)

í **Take the chance of exploiting ESA for YOUR benefits**
(apart from VSCR)

í **Exploit ESA architecture ...**

1. **to SAVE I/Os**

   - follow the concept of DIM
   Refer to the charts on DIM

2. **to SPEED UP I/Os**
   I/Os you cannot save

   - use faster I/O attachments/devices
   Use ESCON, DASD caching, 3390/9345 with dynamic path selection

3. **to MORE OVERLAP I/Os**

   - set up more concurrent partitions/tasks
   E.g. more concurrent batch partitions

í **It is wise to apply these ESA concepts for your own benefits**
Refer to the VSE/ESA 1.3 Exploitation Checklist

VSE/ESA 1.3/1.4 Overall Performance

General

31-bit Virtual Addressing

By support of 31-BIT VIRTUAL ADDRESSING, VSE/ESA 1.3/1.4 provides significant increase of the capacity of a single VSE system and even a single CICS partition. The exploitation of 31-bit virtual addressing is the biggest growth enablement in VSE/ESA 1.3 compared to Release 1 and Release 2 systems, which already provided significant growth potential compared to VSE/SP systems.

Virtual Storage Constraint Relief (VSCR)

VIRTUAL STORAGE CONSTRAINT RELIEF (VSCR) for the private space below 16 MB is provided through the following items above the 16 MB line:

- CICS application programs written in VS/COBOL II and HL-Assembler language
- VSAM buffers (incl. DL/1 index buffers)
- Selected CICS internal areas, such as Table Manager control blocks, Temporary Storage Main, Trace Table, Dynamic Transaction Backout buffers and COMMAREA
- VSE page management tables and other internal areas (moved out of shared space below 16 MB).

Data In Memory (DIM)

'DATA IN MEMORY' (DIM) is a proven concept, introduced with MVS/XA and continued with MVS/ESA, to keep more data in virtual and real storage. By avoiding I/O operations to DASD, DIM provides significant overall response time and throughput improvements. DIM in VSE/ESA 1.3/1.4 is provided by

- the ability to assign huge VSAM buffer areas above the 16 MB line, including up to 15 (CICS) / 16 (Batch) VSAM LSR buffer pools
- CICS Data tables
- Data spaces
- Virtual disk support.

VSE/ESA 1.3/1.4 Overall Performance ...

Further Performance Enhancements

Besides VSCR and DIM, VSE/ESA 1.3/1.4 provides FURTHER PERFORMANCE ENHANCEMENTS, improving batch elapsed times or transaction response times, such as:

- Support of 3990-3/6 DASD Fast Write
- Larger VSAM physical block size
- POWER queue file in partition space.

Workload Specifics

Batch Programs

VSE BATCH PROGRAMS essentially will benefit from reduced numbers of DASD I/Os by using virtual disk for workfiles. Batch throughput increase for such type of jobs will be up to 30% and more.

24-bit CICS Applications

Unchanged 24-BIT CICS APPLICATIONS will benefit by the re-siting of control blocks and, mostly, of VSAM buffers above the 16 MB line.

The relief (increase in available private space for user applications) will be up to about 3 MB and more, depending on the share of data buffers of a customer's workload before VSE/ESA 1.3/1.4.

In case a pre-VSE/ESA 1.3 system is running only with a small total amount of VSAM buffers, the gain in VSCR will be small, but, on the other hand, the potential for saving VSAM I/Os by enlarged VSAM buffers above the line is much higher.

VSE/ESA 1.3/1.4 Overall Performance ...

31-bit CICS Applications

31-BIT CICS APPLICATIONS will effectively no longer be storage constrained by the size of user programs.

This, plus control block and VSAM buffer VSCR, will significantly remove virtual storage as a capacity bottleneck and will allow better exploitation of the processors.

CICS Data Tables

Data Tables offer higher performance access to VSAM data sets by maintaining a copy of the data in memory.

Less Complex CICS setup

By the increased capacity of CICS/VSE for all types of applications, the cost of today's complex communication configurations with several CICS partitions and MRO or ISC will now be recoverable. Thus CPU-time savings in the region of 15 to 40% are expected for such environments.

VSE/ESA 1.3/1.4 Overall Performance ...

Ù Possible CPU-time increase (ITR degradation)

With the introduction of additional ESA features in VSE/ESA 1.3.0 (such as 31-bit addressing or data spaces) the basic VSE pathlengths have been impacted to some extent as compared to VSE/ESA 1.2 or 1.1. This may increase the CPU-time of a CICS transaction or a batch program, and may be different for each customer.

However, considering an 'apples to apples' comparison between 1.3 and 1.1/1.2,

VSE/ESA 1.3 may use around 5% more CPU-time

- depending on environment
- without exploiting new features

CPU-time per transaction can be reduced by:

- reduced frequency of CICS program compressions or short-on-storage conditions,
- reducing the need for CICS MRO by fewer/bigger CICS partitions,
- exploiting more and better tuned VSAM buffer pools,
- using CICS Data Tables.

Bigger VSAM buffers and CICS Data Tables will improve the response times by reducing the number of DASD-I/O operations.

Refer to 'SC33-0703 CICS/VSE 2.2 Performance Guide' for tuning reco-mmendations.

Also, with less or faster I/O operations a potential is given to higher utilize an ES/9000 processor.

í VSE/ESA 1.3/1.4 is a 'Capacity Release'

VSE/ESA 1.3/1.4 Capacity Potential

Potential for higher throughput/capacity

„ **Significant growth potential**

for individual CICS partitions via VSCR

(more terminals/applications/files...)

with more concurrently active CICS/Batch partitions

via more partitions/real storage

í **Higher throughput achievable**

„ **Exploitation of Data in Memory for less I/Os and better Response Times**

„ **Note:**

Residual processor power (storage) mandatory

Without increasing the #terminals or partitions,

naturally, a higher actual throughput can only be achieved

if Online response times better
and faster user reaction

if Batch elapsed times shorter

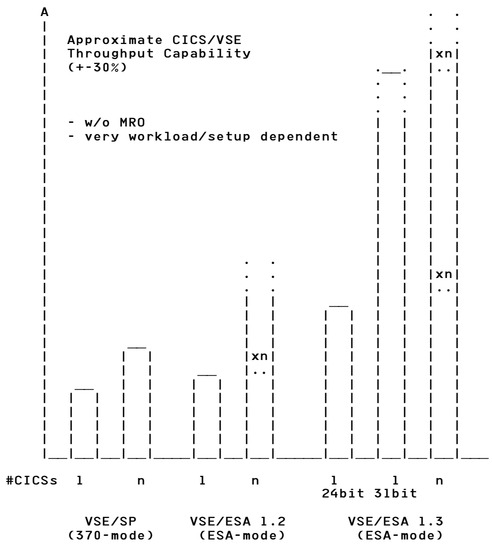
VSE/ESA 1.3 Factors for 'Growth Enablers

Factors for 'Growth Enablers'

Feature	From VSE/SP 4	To VSE/ESA 1.3/1.4	Factor
Real Storage	16MB	256MB+ 2GB (theor)	>16 .. 128(theor)
Total Virt.Storage	128MB	2-3GB 90GB (theor)	>16 .. 700+(theor)
Partition Size below 16M	9 .. 10MB	11 .. 12MB	1.2 .. 1.3
Areas moved above 16M (VSCR)	none	VSAM buffers 31-bit appl. CICS areas System code	-
Data in Memory	no	Larger VSAM buff. & Mult.LSR pools Data Tables Data Spaces & Virtual Disk	-
Number of Address Spaces	9	180 .. 200	about 20
Data Spaces	none	'any'	-
Partitions	12	180 .. 200	about 16
Number of Local Devices	254	1024	4.0
Number of Channels	16	256	16

CICS/VSE Capacities

Rough CICS Capacity Estimates

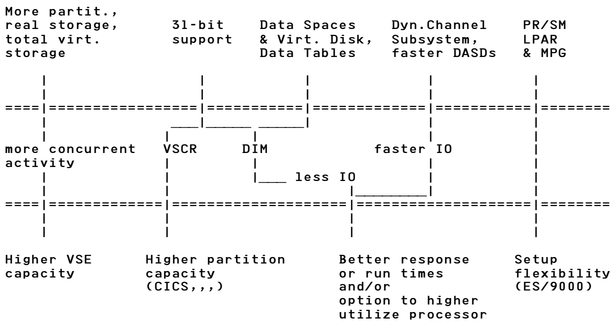


- Multiple CICSs (n) in VSE/SP are 16 MB real bound
- Single CICS 24 bit application capacities are private space bound, even if much more private 24-bit space available in VSE/ESA 1.3 (relief is a function of total VSAM bufferspace before VSE/ESA 1.3).
- To realize these increased throughputs, naturally, the pertinent hardware (processor power, real storage, I/O configuration) is required

Performance Deliverables and Effects

VSE/ESA 1.3 Performance/Capacity

Essential Performance Deliverables



- VSCR = Virtual Storage Constraint Relief
- DIM = Data In Memory

Essential Performance Effects

Performance Deliverables and Effects ...

VSE/ESA 1.3 CPU-time

Essential Performance Impact Factors

```
                                                       Bigger
 New functions,  Virtual               CICS Data       CICS        PR/SM
 addt'l code     Disk       Less IOs   Tables          capacity    MPG

      |              |          |           |           |          |
      |              |          |           |           |          |
  ====|==============|==========|===========|===========|==========|=======
      |              |          |------<----|           |          |
      |              |          |           |           |          |
  Increased      Cost       Savings    Savings      Less        Preferred
  base paths     per        per        per READ     CICS MRO    guest
      |          saved IO   saved IO                TR/FS       support
      |              |          |         |             |          |
      |              |          |         |             |          |
  ====|==============|==========|=========|=============|==========|=======
      |              |          |         |             |          |
      |              |          |         |             |          |

   about         -5%(9221)  up to      up to       up to       better
   +5% CPUT      to +10%    -10%       -20%        -10%/-30%   performance
  (untuned)                                                    if >1 guest
  vs 1.1/1.2     ----- CPUT vs 1.3 untuned -------------      (ES/9000)


  - CPUT   =  CPU-time (per transaction or job)
  - TR/FS  =  Transaction Routing/Function Shipping
```

Essential CPU-time Effects

Clearly distinguish between CPU-time and CPU utilization!

At same CPU-time a faster batch job e.g. must cause higher CPU utilization

VSE/ESA 1.3 Performance Line Items

PART B.

VSE/ESA 1.3 Performance Line Items

Overview

- **General Performance Goals**
- **Performance Line Item Categories**
- **31-bit Exploitation**
- **H/W Support**
- **Further Enhancements**
- **Customer Type Specific Benefits**

General Performance Goals

Performance goals seen from a customer perspective:

Batch jobs

- **CPU-time as small as possible**
- **Fewer/Faster DASD I/Os ... per batch job**

CICS transactions

- **CPU-time as small as possible**
- **Fewer/Faster DASD I/Os ... per tx**
- **Less virtual storage constraints below 16M**
- **Higher CICS capacity for 24 bit**
- **High CICS capacity for 31-bit**

Overall VSE

- **Higher overall VSE capacity**
- **Better exploitation of**
  - processor speed/capacity
  - processor storage
  - I/O resources

Ultimate customer aims

- **Shorter elapsed time for time critical batch jobs**
- **Higher batch throughput with lowest impact on CICS**
- **Better tx response times at given throughput, and/or**
- **Higher tx throughput at given response times**

- **Flexible/effective control and exploitation of H/W and S/W resources**

VSE/ESA 1.3 Performance Line Item Categories

A) 31-bit exploitation

- **Virtual Storage Constraint Relief (VSCR)**
- **Data In Memory (DIM)**

  VSE/ESA 1.3 starts to exploit DIM, which can be viewed as a logical extension of VSCR:

  Moving data buffers above the line: 'VSCR'

  Using bigger/more buffers to save IOs: 'DIM'

VSCR and DIM are provided through:

- 31-bit user applications (VSCR)

- 31-bit buffer areas for files (VSCR, DIM)

- 31-bit functions for system internal purposes (VSCR)

- Further items for more Data In Memory (DIM)

(CICS Data Tables, Data Spaces, Virtual Disk...)

Pre-req for effective VSCR and DIM and further growth:

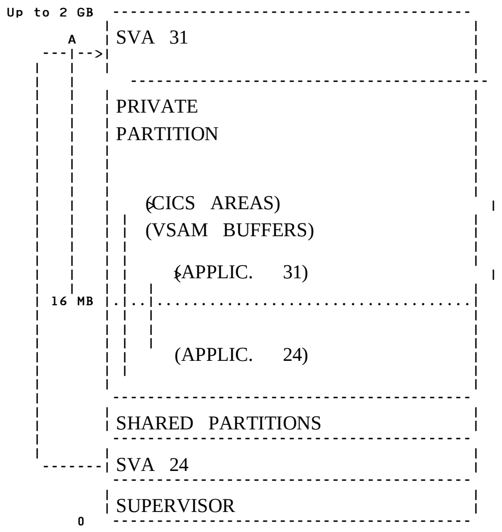
Enough virtual storage (90 GB, potentially)
real storage (2 GB, potentially)

B) Hardware Support

C) Further Performance Enhancements

VSE/ESA 1.3 31-bit Addressing

More Space Available (VSCR and DIM)



A) 31-bit Exploitation

1. 31-bit user applications

„ **Overview**

	COBOL II	High Level Assembler	DL/1 1.10	SQL/DS 3.4 and up	C/370	PL/I
CICS	X	X	X	X	-	LE/VSE
Batch	X	X	X	X	-	LE/VSE

- CICS means CICS command level
- 31-bit batch programs must keep application part with VSAM macros (GET, PUT,...) below the line (RMODE=24), e.g. as subroutines

„ **COBOL II applications (and working storage)**

- COBOL II 31-bit support (Release 4)
- Working storage for COBOL II programs with AMODE=31,DATA=31 (Below the line only 64K available, above up to 128 MB in theory) (If DL/1 used, DATA=24 is required)

„ **High Level Assembler applications**

„ **DL/1 applications (COBOL II, HL-Assembler)**

DL/1 code and data component remain below 16M, refer to the table 'DL/1 31-bit exploitation'

„ **SQL/DS applications (COBOL II, HL-Assembler)**

- New Release 3.4

A) 31-bit Exploitation ...

2. 31-bit buffer areas for files

„ **31-bit VSAM buffers**

Type of application (AMODE)		NSR Index (+Upgrade Set)	NSR + LSR Data Index+Data
Batch	24-bit	X !	-
	31-bit	X	RMODE31=BUFF|ALL
CICS	24-bit	X !	X !
	31-bit	X	X
X = automatically, only partition space above 16M required KSDS, ESDS, VRDS and RRDS			

a) Index and Upgrade set (alternate index) buffers:

(in the average about 1/3rd of all VSAM buffer areas)

May always reside above 16M (CICS and Batch), even for 24-bit applications, since index buffer location is transparent to application. LSR only if also the data reside above 16M, since data and index of an LSR file are always in same LSR pool.

b) Data buffers:

MOVE-mode : records is MOVEd to the application.
LOCATE-mode: only a pointer to LOCATE the record is supplied

Batch, 24-bit-only languages:

- not possible
  Data buffer location only for MOVE-mode transparent to application (where VSAM record is moved from buffer to the application).
  But no means provided to tell VSAM in advance that only MOVE-mode will be used in the RPLs.

Batch, 31-bit capable languages:

- if ACB defined with new parameter RMODE31=BUFF|ALL
  (automatically done by COBOL II)
  (both for MOVE and LOCATE in RPL)

CICS applications:

- CICS makes sure that the ACB is equivalent to RMODE31=BUFF|ALL, no FCT parameter required/available for that. (applications of any language, CICS internally uses MOVE-mode only)

A) 31-bit Exploitation ...

31-bit buffer areas for files ...

„ **Multiple VSAM LSR pools above the line**

CICS 15, Batch 16 pools

KSDS, ESDS, VRDS and RRDS

Much improved data/index differentiation and tuning:

- less CPU-time for buffer searches (if subpool smaller)
- less I/Os (if subpools increased)

„ **DL/1 index component KSDSs buffered above 16M**

MOVE mode instead of LOCATE mode used by DL/1 internally.

Refer to the table 'DL/1 31-bit exploitation'

„ **SQL/DS 3.4 buffers for the data base**

VSAM ESDS with User buffering and CI-processing

Also, internal SQL/DS control blocks above the line.

A) 31-bit Exploitation ...

3. 31-bit functions for system internal purposes

„ **31-bit system routines**
XPCC (9K code), Data space code (37K) and control blocks

„ **31-bit System GETVIS**
System control info and workareas

„ **Page management tables outside of shared space**
Enabling growth of total VSE 'size'

„ **CICS 2.2 areas above the line**
On top of VSAM buffers for user files and RSD

TS main
Used for various functions

Application COMMAREA for all applications
(avg up to 4K/active tx, user defined, 32K max)

When receiving program is 24-bit, a short-living work-COMMAREA is used below the line.

DTB buffers
(500 bytes default per act.task)
(No more spill to TS AUX (DASD) possible, same as in MVS)

Trace table
(default: 125 entries= 4K, shipped: 800 =25K, realistic: up to 2000 =64K)

TMP control blocks

(Table Manager directory segments and hash tables for indexed access to any CICS table, 32 byte per entry. E.g. 500 terminals/500 programs/150 transactions/100 files require 39K)

A) 31-bit Exploitation ...

4. Further items for more Data In Memory

(Primary intent is to avoid physical I/Os to DASD)

„ **Data Tables in CICS 2.2 address space above the line**

VSAM KSDS only

> Refer to separate CICS Data Tables charts

„ **ESA data spaces for any type of 'application' or enabling S/W**

Manifold use to store 'data'.

> I/O reduction and RT improvements

> Constraint avoidance for growing data

> Data integrity improvements

Maintenance of the corresponding data on permanent DASD (if required) must be done by 'application', (as e.g. is with User Maintained CICS Data Tables)

„ **Virtual Disk for work or test files**

VSE/ESA 1.3 implementation is done via FBA concept and data spaces

(Optimally buffered VSAM production files not suited)

No change in application, only in JCL

> Refer to the list of Virtual Disk candidate files

B) Hardware Support

VSE/ESA 1.3 vs VSE/ESA 1.1/1.2

„ **ES/9000 uni-processors in basic ESA-mode**

Including the 9121-x11 models and the 9021-711.
Also the newer 211-based 9221s and the 9672-R11

„ **ESCON CTC through VTAM 3.4**

(up to 3x20km)

„ **ESCON 3174 local terminal attachment (SNA, non-SNA)**

Up to 43 km total (20+20+3), but 3174 controller only up to 3 km from ESCON channel or switch

„ **ESCON Multiple Image Facility (EMIF)**

Sharing of ESCON channels between PR/SM LPARs

Available on all newer ES/9000 processors

„ **3990-3/6 DASD Fast Write / Cache Fast Write**

Support of these extended caching functions.
Error correction code for DFW and CFW available.
Cache Fast Write not exploited by VSE itself.

> Reduction of WRITE I/Os to about 3 msec for cache hits (very similar caching benefits as in MVS)

> Refer to the example for DASD caching benefits

„ **3990-3/6 Enhanced Fast Dual Copy**

Using DFW both for the secondary and primary writes.

Mirror DASD is primarily an availability item

C) Further Performance Enhancements

„ **Larger VSAM physical blocksizes (up to 30K vs 8K)**

3390 track utilization will increase from 87.2% to 98.2%
(CI-size = physical block size = 18K, as an example)

CI-sizes =8K and above

> Faster I/Os for sequential access

> Fewer index I/Os through larger max-CAs

> Less CCWs cause faster CCW translation

„ **POWER queue file in partition if in private space**

16 POWER Queue entries per 4K page

> Saves SVCs for VIO POINTs, no I/O savings, VPOOL requirement reduction.
For POWER queue commands up to 15% CPU-time savings observed.

„ **CICS 2.2 usage of VSAM dataset name sharing**

For VSAM NSR and LSR base and alternate index files, opened multiple times (via multiple ACBs) within the same partition. Mostly an integrity benefit.

> Slightly more efficient buffer usage
or

> Slight virtual storage savings, mostly for VSAM NSR

„ **VSE/ICCF as CICS subtask**

í **Separate startup and shutdown of ICCF**
CICS GETVIS space for ICCF only occupied if ICCF up

í **Slight pathlength savings in CICS systems with ICCF installed**

Compared to elder 'SVC intercept' scheme, when CICS ran in ICCF 'I/A partition 0'

C) Further Performance Enhancements ...

„ **HL-Assembler vs VSE-Assembler**

ESA/390 capability at compile time (31-bit, access registers ...)

Compile step performance:

- Sample results for compile reductions (A-decks):

```
Elapsed time  40 - 50%
CPU-time      10 - 20%
DASD I/Os     50 - 70%
```

- When E-deck conversion required for HL-Assembler:

  Additional workfiles, but addtl files suited for Virtual Disk

„ **LIBR functions extended**

SEARCH member, COPY member into same sublib, BACKUP member

> Increased productivity/performance

„ **VSAM B/R functions extended**

- Backup/Restore from/to Disk (e.g. automatic operation)
- COMPACT/NOCOMPACT parameter for Backup

„ **More I/O devices**

- Up to 1024, requires VTAM 3.4 if used for local non-SNA terminals
- Only possible with VTAM, no support for BTAM anywhere in the system
- With VTAM 3.4 possible already on top of VSE/ESA 1.2.1

> Device number relief for VTAM local non-SNA

VSE/ESA 1.3 Customer Type Specific Benefits

Which benefits for which customers?

(If more virtual storage is used, more real memory is required)

(If DIM exploitation results in more throughput, more CPU resources are required)

Virtual Storage Constraint Relief

Provided to all installations with MODE=ESA.

Especially of benefit for customers with

- exhausted private space below 16M,
  be it by - amount of CICS applications
  - amount of VSAM buffers
  - amount of shared space required
  (e.g. shared VTAM buffers, shared partitions...)
- multiple CICSs, using MRO transaction routing/function shipping for capacity reasons
- intent to combine workloads into a single VSE or CICS
- intent to grow further within a single CICS

Data In Memory Items:

CICS Data Tables

Customers with

- intensive full key VSAM KSDS READs in CICS applications

Multiple VSAM LSR Pools

Customers with

- CICS VSAM KSDSs that do not benefit from CICS Data Tables
- need for better tuning of many VSAM files with various request types

VSE/ESA 1.3 Customer Type Specific Benefits ...

Which benefits for which customers? (cont'd)

Virtual Disk

Customers with

- high DASD I/O time content in batch elapsed times
- many I/Os to 'work' type of files
- test files
- need to better exploit batch window
  when processor power is available for increased throughput
- chances to setup small read-intensive shadow files

  No benefit in Virtual Disk
  - for optimally buffered VSAM files
  - when workfile I/Os are highly blocked or overlapped
  - when not enough real storage is available
  - when no processing power is left

Hardware Support:

3990-3/6 DASD Fast Write

Customers with

- workloads less suited for Data In Memory
- write intensive workloads to permanent files
- no chance of caching VSE Lock File in VM

All ES/9000 uni-processors

Including the new 'x11'models, which also provide EMIF

Customers with

- high growth rate
- need for such powerful processors

CICS Response Time Considerations

Better response times in VSE/ESA 1.3 if

Ù **reduced number of I/O operations achieved by tuning in systems with higher I/O time share in response times today**

- via exploitation of Data In Memory,

  e.g. CICS Data Tables or more/better tuned VSAM LSR or NSR buffers/setup
- via avoidance of paging

Ù **physical DASD-I/Os avoided by usage of Virtual Disk**
mostly beneficial for batch elapsed times

Ù **3990-3/6 Extended Caching Functions (DFW) used in systems with WRITE intensive workloads**

Ù **processors and/or DASDs faster**

Ù **less CPU-time/tx required**

- via less/no more required MRO tx-routing or function shipping
- via avoidance of frequent CICS program compressions and SOS conditions
- via workloads suited for CICS Data Tables

VSE/ESA 1.3 Virtual Storage Constraint Relief

PART C.

VSE/ESA 1.3 Virtual Storage Constraint Relief

Overview

Ù **CICS/VSE Storage Layout**

Ù **SQL/DS 3.4 Performance Items**

Ù **Page Management Tables**

Ù **Supervisor Comparison**

Ù **Shared Space Summary**

CICS/VSE 2.2 Storage Layout

```
30MB ---> ----------------------------------------------
 (e.g.)  | -------   --------    ---------------------  |  |
         ||       | | Data   |  |                     | |  |
         ||  TS   | | Table  |  |       D A T A       | |  |
         || Main  | | Hash.  |  |     T A B L E S     | |  |
         ||       | | Tables |  |                *    | |  |
         | -------  --------    ---------------------   |  |
         | -------    ------    ---------------------   |  |
         || DTB   |  |Trace |  |    VSAM-Buffers     | |  | GETVIS
         ||Buffers|  |Table |  |  including Restart  | |  | LOC=ANY
         ||       |  |    * |  |  Dataset DFHRSD   * | |  |
         | -------    ------    ---------------------   |  |
         |                                              |  |
         |  ---------------------     ----------        |  |
         | |  31-bit-Applications |   | Table    |      |  |
         | |                      |   | Manager  |      |  |
         | |     RMODE(ANY)     * |   | Areas  * |      |  |
         |  ---------------------     ----------        |  |
         |    ---------------------      -------------      |  |
         | |  Working Storage     |   | All          |  |  |
         | | VS COBOL II DATA(31) |   | COMMAREAs    |  |  |
         |    ---------------------      -------------      |  |
        /                                              /  /
         |                GETVIS 31-Bit                 |  |
 16MB -->|- - - - - - - - - - - - - - - - - - - - - - - |  |
         |                                              |  |  A
         |                                              |  |  |
        /                                   --------   /  /  /
         |   --------------------           |VSAM     | |  |  | GETVIS
         | | ICCF with Interac- |          |Control | |  | LOC=BELOW
         | | tive Partitions    |          |Blocks  | |  |  |
         |   --------------------            --------  |  |  |
         |                GETVIS 24-Bit                 |  V  |
         |- - - - - - - - - - - - - - - - - - - - - - - |     -
         |                                              |  A
         |           CICS NUCLEUS and DSA               |  |
        /                                              /  /
         |   --------------------------       ---------      |  |
         | |  24-bit-                    |   |Active   |  |  | DFHSIP SIZE
         | |  Applications               |   |24-Bit   |  |  | Program
         | |       RMODE(24)             |   |COMMAREAs|  |  | Area
 e.g.    |   --------------------------       ---------      |  |
 5MB ---> -------------------------------------------   -

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


SQL/DS 3.4 Performance Items

„ **Support of 31-bit applications**
COBOL II, HL Assembler running under CICS/VSE 2.2

„ **Data base buffers above the line**

„ **Higher number of NDIRBUF, NPAGBUF feasible**

„ **Selected internal SQL/DS control blocks above the line**

„ **Higher number of concurrent transactions supported (NCUSER)**

í **Up to roughly 85 MB virtual partition size**
Limited by available real storage and checkpoint duration

„ **Support of multiple data bases or SQL/DS partitions**

„ **Usage of Virtual Disk for internal DBSPACEs**
Better elapsed times for batch and maintenance jobs with large sorts and joins

„ **Reduced logging**
for cases where a single phase commit suffices

„ **ARIPRDI phase now SVA-31 eligible**
Required for all SQL/DS applications

„ **Tuning information in new SQL/DS manual**
'SQL/DS Performance Tuning Handbook', SH09-8111

VSE/ESA Page Management Tables

„ **All VSE/ESA page mgmt tables outside shared space**

Page and segment tables (1.3K per MB) in extra private address space
Page frame table (8K per MB) in real storage

Table Space Requirements

(MODE=ESA, approximate and average values)

VSE/ESA 1.3 real storage for page mgmt tables:

1.3 KB per MB VSIZE + 8 KB per MB real

VSE/ESA 1.1/1.2 shared space for page mgmt tables:

1.3 KB per MB VSIZE + 4.5 KB per MB real

Examples for Table Space Requirements

Approximate values:

	Medium	Larger	
MB virtual (VSIZE) MB real	80M 32M	256M 64M	
VSE/ESA 1.1/1.2	250K	624K	Shared space, PFIXed
VSE/ESA 1.3	360K	848K	Real storage and Private space, PFIXed

í **Gaining n KB of shared space (vs VSE/ESA 1.2) by spending 0.3xn KB additional real storage**

í **Enabling significant growth of VSE/ESA storage/systems**

VSE/ESA 1.3 Supervisor Comparison

Supervisor Comparison

	VSE/ESA 1.2.2	VSE/ESA 1.3.0	Delta
Supervisor V-SIZE	856K	516K	-340K
- Page mgmt tables	392K	0K	
= Net virtual size	464K	516K	+52K

Net supervisor increased by about 50K

í **Savings in total supervisor size = 340K**

```
Supervisor savings higher, if VSIZE, RSIZE higher

(In example above for VSE/ESA 1.2.2:

    VSIZE = 80 MB, RSIZE = 64 MB, only)


VSE/ESA 1.3 total supervisor size still depends on:

            - generation options
            - SYS BUFSIZE, CHANQ, SDSIZE  parameters
```


VSE/ESA 1.3 SDL and VLA Comparison

SDL and VLA Comparison

```
(Standard system)
                             VSE/ESA       VSE/ESA      Delta
                              1.2.2         1.3.0

Number of entries used         243           261        + 18 entries

SVA (24) Used Space           1325K         1458K       + 133K
SVA (31) Used Space             -            37K


More details:


Some new phases                          $IJBHDUP   6K
(loaded low)                             $IJBALET   2K
                                         $IJBSSM1   3K
                                         $IJBVDII   6K

Some increased phases                    $IJBAR    +13K
(loaded low)                             $IJBLBR   +18K
                                         $IJBSDMP  +36K
                                         $IJBMAP   + 5K
                    VTAM                 IST...    +12K   (193K to 205K)
                    VSAM                 IKQ...    +18K   (402K to 420K)
                    Misc                 .......   +23K

Moved to high                            $IJBSXPC  -9K (XPCC)
                                         _____________
                                                  +133K

Phases (loaded high)                     $IJBALE    7K
                                         $IJBALET   2K
                                         $IJBCVT    2K
                                         $IJBDSP   |
                                         $IJBDSPA  |7K
                                         $IJBDSPD  |
                                         $IJBSDSP   6K
                                         $IJBSXPC   9K
                                         $IJBVDIC   6K
                                         _____________
                                                   37K

Note: $IJBSDMP includes new macros for data space dumps
      and code/area for DUMP reentrancy.
```


VSE/ESA 1.3 Space Summary

Available partition space below 16MB

```
Observations from sample measurements (Figures will vary with workload and
product configuration/setup)
(VSE/ESA 1.3.0 vs VSE/ESA 1.2.2)
RAMP-C, VSAM NSR, no products with additional shared storage needs
```

„ **Total size of required shared space below the line**

```
                          | Delta  |   ESA 1.2.2 -->  ESA 1.3.0
 _____________________________________________________________
 Supervisor               | -340K  |    ( 856K   -->    516K )
 SDL list                 |    0K  |    (  32K   -->     32K )
 Virtual Library Area     | +133K  |    (1325K   -->   1458K )
 System GETVIS24 (used)   | + 76K* |    (1068K   -->   1144K*)
 Label Work Area          |    0K  |    ( 108K   -->    108K )
 VPOOL                    |    0K  |    (  64K   -->     64K )
 Shared partitions        |    0K  |    (   0K   -->      0K )
 _____________________________________________________________
 Total Shared Used        | -131K  |    (3453K   -->   3322K )
                          |        |
 Rest                     |  **    |    ( 643K   -->    774K )
 _____________________________________________________________
 Total Shared             |    0K  |   ( 4096K   -->   4096K )

 - Refer to charts for Supervisor and VLA details

 * Note that due to more conservative determination of the available
   space in the AR GETVIS display, VSE/ESA 1.3 values may slightly
   be lower than for earlier releases (Also those areas are now
   counted as used, which already belong to reserved subpools, but
   still are available).

** Space required for 1M segment rounding, available for peaks.
   Segment rounding space is not lost, if e.g. CICS phases are
   loaded into the (increased) SVA/VLA.
```

í **Available private space below the line for partition allocation:**

12 MB (this example)

```
Formally, in this example, the same private space is available
for the maximum size of a partition.

Refer to the next page for the private space available WITHIN
a CICS partition.
```


VSE/ESA 1.3 Space Summary ...

Available space below 16MB (cont'd)

„ **Private space within CICS partition below the line**

```
                     | Delta  |   ESA 1.2.2 -->  ESA 1.3.0
 __________________________________________________________
 CICS nucleus        |  +60K  |   ( 728K   -->    788K )
 CICS DSA used       | -  xK  |   (    K   -->       K )
 Partition GETVIS24  |-4104K  |   (5028K   -->    924K )
               used  |        |
 __________________________________________________________
 Total used          | - 4M   |       -      -->     -

 - CICS DSA usage via CICSPARS (or CICS page allocation map),
   xK delta is moved above the line, depending on workload
   (Refer to calculation on separate chart)


 - Gain in used Partition GETVIS mostly by VSAM buffers relief

   Here, about 4 MB more private space is available
   within the total CICS partition of same size.
```

í **The gain in private space (VSCR) below the line results mostly from VSAM buffers, being moved above (24 bit applications)**

If the total VSAM buffer space was small

VSCR is smaller, ... but

chance for reducing I/Os by bigger buffers above the line is bigger (VSE ran 'de-tuned' regarding CPU-time and IOs)

VSE/VSAM Control Blocks Below 16M

In spite of most VSAM buffers being able to reside above the line, ...

VSAM Control Blocks in 24-bit Partition GETVIS

- **14K per open catalog**
  Control blocks and basic buffers
- **50K during OPEN and CLOSE**
  OPEN control blocks and catalog check routines
- **Miscellaneous VSAM control blocks**
  ACBs, ARDBs, OALs, OPNWAs, AMBLs, AMDBs, ...
- **Add'tl control blocks for AIX and PATH**
- **File related control blocks (per file)**

	Base		Add-on
	NSR	LSR	NSR or LSR
No PATH	7K	3.25K	(STRNO-1) x 1K
PATH	9K	5.25K	(STRNO-1) x 2K

- **NSR related control blocks (PLH, 1 per NSR-string)**
  572 byte + max key_length (keyed files)
  544 byte (non-keyed files)
- **LSR related control blocks**

  (includes PLHs for LSR files)

  = 2128 + 72 x #subpools + 108 x #buffers_in_pool
  + (920 + max_key_length) x STRNO ...per LSR pool

- **With new opportunities for more buffers...
  keep in mind the space impact of STRNO**

For more details refer to 'VSE/VSAM Users Guide'

Virtual Storage Constraint Relief (Summary)

Virtual Storage Constraint Relief (Summary)

- **VSCR for more/bigger applications and/or higher tx rate**
  - **12 MB private space (partition size) below the line**
    by moving areas out of shared space below 16M
  - **Up to 3 MB and more space available within partition below 16M**
    by moving VSAM buffers for 24-bit applications high, apart from CICS areas

    **If the total VSAM buffer space was small,
    VSCR is smaller, ... but
    chance for reducing I/Os by bigger buffers above the line is bigger
    (VSE ran 'de-tuned' regarding I/Os and CPU-time)**
- **Especially of benefit for customers with**
  - **exhausted private space below 16M**
  - **multiple (connected) CICSs, using MRO transaction routing/function shipping for capacity reasons**
  - **intent to combine workloads into a single or fewer VSE or CICS**
  - **intent to grow further within a single CICS**

VSE/ESA 1.3 Data In Memory

PART D.

VSE/ESA 1.3 Data In Memory

Overview

- **Data in Memory Aspects**
- **CICS/VSE 2.2 Data Tables**
- **VSAM Multiple LSR Pools**
- **VSAM NSR with bigger buffers**
- **DIM Experiments (RAMP-C)**
- **DIM Examples**
- **Virtual Disk Benefit Areas**
- **Virtual Disk Results**
- **Virtual Usage Hints**

Data in Memory (DIM)

Rationale

Physical I/Os to DASD last very long (vs processor speed)
(1 week (uncached) or 1 day (cached) vs 1 sec)
- **Avoid I/Os whenever possible
  by storing more data in virtual storage (--> DIM)**

I/Os that can be avoided

For permanent data required on DASD:

READ I/Os
(immediate updates to disk still required)

For temporary/work data so far on DASD:

READ and WRITE I/Os

Direct Benefits

- **Improved Response Time through less I/Os**
  I/O reductions depend on tuning degree and workload
- **Reduction of overall required 'short living' areas**
  Faster response times give some VSCR in CICS
- **Smaller CPU-time per tx, through**
  depending on tuning degree and workload
  - **less I/O-supervisor activities**
    (less CCW setup/translations, less I/O interrupts)
    (But: VSE Virtual Disk increases supervisor pathlength)
  - **fewer processor cycles for 9221 I/O u-code**
    (all 9221s w/o separate I/O processor: dyadics, non-211 based unis)

Important

- **Provide increased real storage**
  - **to allow effective use of additional virtual storage, i.e. to avoid exchanging file-IO with page-I/O**

Data in Memory (DIM) ...

Data in Memory - Cost vs. Benefits -

- **Cost**
  - **Processor (Central) Storage**
- **Overall Benefits**
  - **Less I/Os and CPU-time per tx (direct benefits)**
  - **Increase of user productivity**
    from better response times
  - **Savings in system programmers time**
    from less IO tuning
  - **Savings in I/O configuration**
    by reduced I/O rate (at same throughput):

    Less channels, control units, DASD arms required for acceptable I/O response times
    (Same DASD space required for files, slight increase for the PDS)

    and/or

    better I/O times
    at same DASD configuration
  - **Increased capacity of the entire system**
    due to reduced CPU-time per transaction

    and/or

    due to driving the CPU harder for same response time,

    - compensating part of the response time improvements
    - postponing the need for update/replacement

Data In Memory Exploitation

Components for Effective DIM Exploitation

1. **Virtual Storage**
   - **31-bit Addressing**
     - **31-bit Components**
     - **31-bit Data Areas**
     - **31-bit User Applications**
   - **Multiple Address Spaces & Dynamic Partitions**
   - **Data Spaces incl. Virtual Disk**
2. **Real Storage**
3. **Virtual Storage Constraint Relief**

CICS/VSE 2.2 Data Tables

CICS Data Tables

A CICS/MVS 2.1.1 feature, standard in CICS/ESA 3.1.

- **CICS data 'look-aside' within CICS address space for full-key READs of VSAM LSR defined KSDS files**
  CICS does a direct look up, based on the key, VSAM is not aware

  Normal VSAM LSR services required:
  - if full key READ record not in table
  - for sequential or generic reads (e.g. BROWSE)
  - for UPDATE requests
- **2 table types:**
  Defined with FCT TYPE=CICSTABLE|USERTABLE

  CICS Maintained Tables (CMT)
  CICS maintains integrity of data between table and source file.

  User Maintained Tables (UMT)
  Update of source file only controlled by application. Several file operations not supported, application changes may be required
- **Table loaded at file OPEN time**

  Mostly OPEN at CICS startup: FILSTAT=(,OPENED)

  Sometimes opened on first reference: FILSTAT=(,CLOSED)

  1 table (default SIZE=100 000 records) per file

  All or selected parts of a file (controllable by user exits)

CICS/VSE 2.2 Data Tables ...

- **Performance benefits**

  Fast, efficient access to data, by

  Pathlength to access records via table shorter than access via VSAM
  Less DASD IOs, if buffer space and real storage additionally available
  Reduction of time of residency for transactions can also give a marked reduction in DSA use

  The last 2 items are common to all usages of DIM

- **Improved transaction response times and throughput, depending on workload**

- **System programmer hints**

  Use CICS data tables in systems with many full-key READs, before increasing VSAM LSR subpools beyond say 300 to 400.

  Use Data CI-sizes as big as possible for data tables.
  Will improve data table load time.

CICS/VSE 2.2 Data Tables ...

Data Tables Performance Observations

„ **Simple Query benchmark**

- Read a key from terminal
- Retrieve VSAM record from source data set (200K records, 40 bytes each)
- Display record on terminal

„ **VSE/ESA 1.3.0 on ES/9121-320**

Comparing 2 different alternatives with DIM:

	Single VSAM LSR pool (8.1 MB)	CICS Data Table (8.5 MB)
Transact/sec	about 80	about 80
CPU utilization	26%	13%
ITR (tx/CPU-sec)	about 300	about 600
Here, no I/Os were saved, since both cases used DIM		

í **Internal Throughput Rate (ITR) was doubled (improved by 100%)**
for such a specific transaction workload (optimally suited for CICS data tables)

í **ITR improvements for other type of transactions, naturally, will be smaller**
(even if also I/Os are saved)

VSE/VSAM Multiple LSR Pools

Benefits of Multiple LSR pools (above the line)

„ **Separation of files with same CI sizes (data, index)**
Such files may be 'unfriendly' to each other (stealing buffers, dominating a subpool during BROWSE etc...)
í **Group/Separate files by application, work shift, usage frequency etc.**

„ **Separation of data and index-CIs (of different files)**
Assure data-CIs do not compete with index-CIs of another file in the same subpool

„ **Full freedom to select optimal data and index-CI sizes**
Without regarding CI-sizes of other LSR files

„ **Shorter subpools for faster searches possible**
Avoid long subpool searches (CPU-time) with low chance of hit ratio increase
- if shared with too many other files
- if already long enough for a single file

í **More subpools with less buffers per subpool use less CPU-cycles**

„ **More than 255 strings possible in total**

Overall:

í **Reduction of VSAM IOs (same subpool sizes)**

or

í **Even higher reduction of VSAM IOs (larger subpool sizes) at cost of some CPU-time**

VSE/VSAM Multiple LSR Pools ...

Tuning of Multiple LSR pools

(DFHFCT TYPE=SHRCTL and BLDVRP definitions)

Independent of the LSR buffers being able to reside above the line...

í **VSAM control blocks remain below 16M**
Refer to the chart on VSAM control blocks in 24-bit GETVIS.

In order to economically exploit the space below 16M...

„ **Define only as many LSR pools as required**

„ **STRNO: Do not largely oversize STRNOs**

For a single LSR pool, the maximum STRNO was and remains 255. At same workload, you may require for the sum of all your pool STRNOs only a few more (since you share them less), as you had before for a single LSR pool.

„ **KEYLEN: Specify correctly**

If you specify that as real maximum of all the KSDS files in the pertinent pool, CICS need not to determine that value at CICS startup time.

„ **Use provided CICS 2.2 LSR Hit Ratio statistics**
with I/Os and #hits per CI-size (subpool) per pool
to optimize #buffers per subpool

- Increase number, until hit ratio does no more increase sensibly
- Decrease number, until hit ratio does no more decrease sensibly

VSE/VSAM Multiple LSR Pools ...

Reserving a subpool for a single file

This may be considered for very specific tuning situations or special treatment/isolation of an individual file.

It may also be a temporary step, just to find out how a specific file behaves on its own, before sharing a subpool.

The benefits or disadvantages of not sharing a subpool with other files vs having the file defined as NSR are:

VSAM NSR vs specific LSR definition of a file

	VSAM NSR	VSAM LSR with reserved subpools for a single file
Buffer Lookaside	No, each file request looks for data only in the buffers for the own string	Yes, lookaside to other strings in same subpool, at cost of some CPU-time
Read integrity for mult. OPENs	No	Yes, if in same subpool
Read ahead for BROWSE	Yes, SHROPT 1 - 3*	No
* Do not use SHROPT 3		

í **Reserving a subpool for a single file: may be performance-wise beneficial.**

This includes BROWSE intensive files, except Read ahead is required.

VSE/VSAM Multiple LSR Pools ...

Multiple LSR Pool Performance Observation

„ **CICS DSW online transaction workload**

„ **ES/9121-320, 430 terminals**

„ **Results**
VSE/ESA 1.3.0, Multiple vs single LSR pool

12 vs 1 LSR subpool, in total 2.6 vs 0.8 MB total buffer space

	Delta
Total LSR hit ratio	+ 20%
# DASD-I/O per tx	- 30%
ITR (tx/CPU-sec)	+ 10%

í **DASD-I/Os reduced by about 30%**

í **ITR improved by about 10%**

Careful selection and setup of multiple pools for 18 files

ITR improvement by saved I/Os bigger on a 9221

VSE/VSAM NSR with Bigger Buffers

NSR Performance Observation

„ **CICS RAMP-C workload with VSAM NSR buffers**

„ **ES/9221-150, 200 terminals**

„ **Results**

VSE/ESA 1.3 vs VSE/ESA 1.2

	Same NSR buffers	1.4 MB more NSR buffers (3 vs 1.6 MB)
DASD-IO/tx	+-0%	-13%
ITR	- 3.7%	+6.3%
Response time	+-0%	-32%

í **About 10% ITR gain within VSE/ESA 1.3 by more VSAM NSR buffers**
Benefits by more VSAM buffers depend on how much the base system is constrained in buffer size

Data In Memory Variations

DIM Experiments with an Online Workload

- CICS RAMP-C workload, 300 terminals (highly active), with varying VSAM setup
- VSE/ESA 1.3.1 in ESA-mode, up to >200 MB CICS partition size
- 9221-170 processor, 256 MB real storage

	CICS GETVIS (MB)	tx/h at 70%	ITR ratio	SSCH /tx	Avg. RT (sec)
BASE: NSR	7	56146	1.00	16.2	0.86
Single LSR, untuned	4	50400	0.90	14.0	1.00
Single LSR, diff. CIs	5	55138	0.98	12.4	0.92
Mult. LSR (15 pools)	16	73836	1.31	7.2	0.57
Mult. LSR, diff.CIs	31	76356	1.36	6.6	0.55
Mult. LSR, data tables	176	78019	1.39	6.6	0.54
Mult. LSR, data tables +diff. CIs	192	79732	1.42	6.4	0.51

- diff. CIs: more different CI-sizes selected for more (and thus shorter) subpools
- data tables: User maintained CICS data tables used for read-only files
- Results gained in controlled environments, not necessarily comparable to other RAMP-C runs. Results in other environments may deviate.

í **Up to 42% more Internal Throughput by DIM**
Corresponds to -30% CPU-time/tx

í **Reduction of I/Os per tx to about half**

í **The more DIM, the better (if setup intelligently)**

„ **Best 'return of investment' with about twice the total VSAM bufferspace:**
+31% ITR, -56% SSCHs, -34% reduced response time

- On 9121s, CPU-time savings would be smaller, RT benefits similar
- More details are documented in: 'Exploiting VSE/ESA 1.3: 40% more RAMP-C Throughput'

Data In Memory (DIM) Example

Effects on Response Time

Take a sample transaction on an ES/9121-210 with 20 I/Os to DASD. Assume a processor online utilization of 70%, resulting in a response time of 0.54 sec:

CPU-time	CPU queue-ing time	DASD-I/O time
40 msec	93 msec	400 msec

Response time = 0.54 sec (17.5 tx/sec, at 70% CPU)

Using data in memory techniques, the number of I/Os can be reduced by say 30%, and the CPU-time per transaction by say about 5% (by CICS Data Tables or bigger VSAM buffers). CPU utilization results in 66%, with smaller CPU queueing time:

CPU-time	CPU queue-ing time	DASD-I/O time
38 msec	77 msec	280 msec

Response time = 0.40 sec (17.5 tx/sec at 66% CPU)

Driving the CPU harder, e.g. to 80%, increases CPU queueing time, but still gives better response time

CPU-time	CPU queue-ing time	DASD-I/O time
38 msec	152 msec	280 msec

Response time = 0.47 sec (21.0 tx/sec at 80% CPU)

Data In Memory (DIM) Example ...

Effects on Capacity

Data in memory saves I/Os and thus also CPU-time, which is a more effective use of the CPU, with potential for increased throughput.

In the VSE/ESA examples of the previous page the 9121-210 handled about 17.5 tx/sec. With 5% CPU-time reduction at same CPU utilization this results in 18.4 tx/sec:

5% more transactions because of CPU-time savings

Increasing the CPU utilization from 70% to 80% means 21.0 tx/sec with improved response times:

20% more capacity for online transactions

with

13% reduction in response time.

Note:
Reduced transaction response times and thus higher capacity at higher CPU utilization can also be achieved by using faster I/O attachments, such as cached devices.

VSE/ESA Data Spaces and Virtual Disk

Performance Benefits of ESA Data Spaces

Any program can work with data in data spaces via Assembler macros,

so do many 3rd party applications.

Improved performance by sharing data more effectively across partitions
- **Less communication overhead (e.g. via XPCC)**
- **Data only at one place**

Exploitation of processor storage

Constraint Avoidance (Growth Potential)
Data which cannot/should not reside in user address space

Especially for VIRTUAL DISK usage of data spaces:

Improved performance if DASD I/Os reduced (additional processor storage mandatory)
- **Less I/O through more data in memory**
  Write and Read I/Os to work or test files

  → **Improved batch elapsed times**
  Highest benefits in single batch jobs
  → **Improved transaction response times**
  if part of response time was DASD-I/O and now is saved

Virtual Disk Benefit Areas

Basic Effect

Avoid physical I/Os to any type of 'workfile'

Basic trade-off:

- I/O savings potentially improve batch elapsed or individual transaction response time,
- at cost of some CPU-time, processor storage and the risk to introduce paging for paging sensitive partitions (CICS, VTAM, SQL/DS)

Conditions for Benefits

Virtual Disk is beneficial in cases where ALL of the following conditions are fulfilled:

1. **Data can be stored in volatile storage**
   - work or test data
   - easily recreatable data
   - data backed up on real disk (read only or read intensive)
2. **Physical I/Os can be avoided and bring run-time benefits**
   E.g. low buffered files, when run-time include I/O times, and where workfile I/Os are mostly non-overlapped with CPU-time or other I/Os
   Alternatively, files on high contention volumes
3. **Additional processor capacity is available**
   - for pathlength increase through I/O intercepts
   - for an increased throughput
4. **Enough processor storage still is available,**
   not to exchange file-I/O with paging, especially when run concurrently with CICS production (with or w/o SQL/DS)

Refer also to:
'VSE/ESA 1.3 Virtual Disk as an Additional Tuning Opportunity'

CPU-time Impact for Virtual Disk

S/W pathlength always increases
(vs. real FBA volume)

→ **in general increased CPU-time**

CPU-time only reduces under specific optimal conditions:

1. No paging at all
AND
2. a) Physical I/Os saved on a 9221
if w/o separate I/O processor
OR
b) more effective blocking vs CKD
OR
c) file on real DASD was unassisted (VM/VSE)

- No paging:
  a) VD usage does not cause to page other pages out
  b) VD pages are not paged out and thus do not have to be paged in later
  (in somehow constrained systems, VD pages will be paged out if not used for some time)
- Saved physical I/Os:
  Only on a 9221 the I/O u-code runs on the same processor as the S/W. Thus only on a 9221 saved physical I/Os will save CPU-cycles.
- More effective blocking
  By this, I/O paths are hit less often and e.g. CCW translations and I/O interrupt code are saved in the VSE supervisor, e.g. SYSLNK FBA vs CKD.

VSE/ESA 1.3 Virtual Disk Performance Benefits

About 30% throughput increase in the example below

Assumptions

- Compile type of job, run in single batch
- 12.5 sec CPU-time of the total job on a 9221-150
- 4000 I/O operations to disk, un-overlapped, with about 30% to work areas (as an example)
- 20 msec average per DASD I/O
- Additional real storage available
  i.e. no exchange of file I/O with paging I/O
  VD pages not paged out

Calculation

- Elapsed Time ET = CPUT + (un-overlapped) IO
- Elapsed time without virtual disk:

```
CPUT =   12.5 sec
  IO = 4000 x 20 msec = 80 sec    ---> ET = 92.5 sec
```

- Elapsed time with virtual disk:

```
    CPUT = about 13.5 sec  (some increase)
      IO = 0.7 x 4000 x 20 msec = 56 sec
---> ET = 69.5 sec  (25% ET reduction = 33% throughput increase)
```

Sensitivity Factors

```
Percentage higher if e.g. - more I/Os to DASD work areas
                          - removing an I/O bottleneck
                          - processor faster
Percentage lower  if e.g. - workload less I/O dependent
                            (I/Os higher blocked or overlapped)
                          - workfile I/Os more overlapped
                          - more partitions active / concurrency
```


VSE/ESA 1.3 Some Virtual Disk Results

Workload/Environment

- **DOS/VS COBOL Compile and Link**
  1233 lines of source code, SIZE=500K (Compile) 100K (Link)
  CBL BUFSIZE=8192
- **ES/9221-150 in ESA-mode, 3380 system volumes, different user volumes**
  Only 'real' workfiles placed on virtual disks:
  SYSLNK 2 MB, IJSYS001-4 4 MB each
  - **Example for single batch environment**

Measured Reductions

9221 H/W monitor and VSE JA used, no paging --> optimal conditions

```
                   CKD-3380     Virt. Disk     FBA-9336-20

Elapsed Time           -60%             -30%
                 (33 sec) -----> (14 sec) <------ (20 sec)

#DASD I/Os             -65%             -45%
                 (1600)   ----->  (570)   <------  (1050)

CPU-time               -14%              -2%
               (4.8 sec) -----> (4.1 sec) <----- (4.2 sec)

CPU utilization
                 (14%)   ------>   (29%)   <-----   (21%)
```

Notes:
- VSE virtual disk has bigger supervisor pathlengths than real disk (3% more total and 10% more supervisor instructions vs real FBA)
- Savings in CPU-time vs FBA-9336 are due to fewer physical I/Os (saved 9221 I/O u-code, no u-code cycle savings on a 9121)
- The higher savings in #DASD I/Os and CPU-time vs CKD-3380 are mostly due to better SYSLNK blocking on FBA
- An increased CPU utilization is a positive effect here, since the CPU-time per job does not increase.

VSE/ESA 1.3 Some Virtual Disk Results ...

Workload/Environment

- **VSE Standard Batch Workload (PACEX) in Single Batch**
  7 jobs: PAYROLL, BILLING, STOCK, COBOL, DL/1, SORT, FORTRAN
  Not especially tuned for I/O savings
- **ES/9221-150, VSE/ESA 1.3.0 in ESA-mode, 3380 system volumes, 2 user volumes per partition**
  All user volumes placed on virtual disks:
  Test environment like, not representative for other environments
  - **Example for maximum potential**

Measured Reductions

(9221 H/W monitor and VSE JA used, no paging --> optimal conditions)

	CKD-3380 user vols	Virt. Disk	Reduction
Elapsed Time	382.9 sec	92.7 sec	-75%
CPU-time	56.36 sec	53.32 sec	-5.4%
#instructions	-	-	+5.6%
# I/Os	21365	2004	-90%
CPU utilization	14.7%	57.5%	-

- All I/Os to the user volumes (90%) were saved, as expected
- Job sequence was executed about 4 times faster
- About 5% of CPU-time was saved, though about 5% more VSE instructions had to be executed on top for VD.

  Savings came from saved I/O u-code on the 9221.
  On a 9121, additional CPU-time would have been required.

VSE/ESA 1.3 Some Virtual Disk Results ...

Workload/Environment

- **Various I/O intensive test jobs (19) in single batch**
  - Assemble and catalog various CICS tables
  - Compile and catalog COBOL II programs
  - VSAM KSDS and LISTCAT test jobs
- **ES/9221-150, VSE/ESA 1.3.1 in ESA-mode, 9336 DASDs**
  1 user volume (SYSWK5) placed on real or virtual disk, containing SYS00x, SYSLNK, VSAM KSDS and catalog

  Test environment, no paging --> optimal conditions

  CPU-times from EXPLORE/VSE 'Flashback Report'

  Runs done by R.Irving, Dallas
- **Measurement Results**

	FBA-9336-20 user volume	Virt. Disk	Delta
Elapsed Time	1337 sec	642 sec	-52%
CPU-time	191.12 sec	205.27 sec	+7.5%
CPU utilization	14.3%	32.0%	-

> Job sequence was executed about 2 times faster

at cost of about 7% CPU-time in the average, with the following variations:

```
CICS table assemblies  + 6%
COBOL II compiles      +1.5%
VSAM KSDS              +10%  (very I/O intensive)
```

> All I/Os to the user volume were saved, as expected

> Here, the savings in 9221 microcode CPU-time could not compensate the increase in S/W pathlength

VSE/ESA 1.3 Some Virtual Disk Results ...

VSE/ESA Label Area on Virtual Disk (VSE/ESA 1.3.1)

- **Some measured Elapsed Time (ET) improvements**

(DLA on Virtual Disk vs DLA on 3380, 9221-150)

Label Area Intensiveness	Activity	ET Reduction
very low	Total batch jobs, few files in system and in JCL	-1%
higher	Job Control only, few files in system and in JCL	-20%
average	CICS startup, some files as job labels	-15%
high	CICS startup, some files, but many std labels *	-50%
* CICS file label searches had to scan about 500 other labels in std label area before		

- Values shown hold for single partition activity
- Elapsed Time reduction the bigger, the higher the number of DLA I/Os to real disk
- Reduction of DLA file contention if many partitions run concurrently
- CPU-time per job may increase, depending on environment

- **Definition via new VDISK parameter**

```
// VDISK UNIT=cuu,BLKS=n,USAGE=DLA
```

Use in $0JCL, best before // EXEC PROC=STDLABEL:

- at creation of 'virtual label area', all labels from the real label area are copied into the virtual disk
- newly added labels are no more written/required on real disk
- (small) real label area only required during IPL
- virtual label area must be kept, cannot be changed in size

VSE/ESA 1.3 Virtual Disk Exploitation Checklist

VSE/ESA 1.3 Virtual Disk Opportunities

Assembly work areas	IJSYS01 to IJSYS03
Compiler work areas	IJSYS01 to IJSYS07
Linkedit work areas (Option CATAL)	SYSLNK
Sort work areas (if req'd)	SORTWK1 to SORTWK9
DTSANALS work areas	IJSYS02
CSP work files	CSP.USER.WORK VSAM
Test/work SAM and VSAM files	
Test/work libraries	
Reproducible input/output files across jobs/job steps e.g. for - batch production - DL/1 reorganization - NCP generation	SYSIPT, SYSPCH
Read only user files	Copied from real disk
Read only libraries e.g. for - PSF fonts - RPG CICS applications - LE read/only modules	Copied from real disk Cannot be made resident Speed up Linkedit (1.4)
Read intensive user files	Copied from real disk, real disk updates by applic.
Read intensive libraries	Copied from real disk real disk updates by applic. (or system programmer)

Above items hold both for BATCH and Online, but Online benefits are smaller since response times benefit less from saved I/Os due to overlap of I/Os and multithread in CICS.

Moving well buffered VSAM files to virtual disk may not be beneficial. The I/Os are already mostly minimized. If lower buffering is used, more CPU-time may be required in I/O paths and their interceptions.

Apart from the read intensive items, all opportunities above do NOT require any change in the application itself. You only have to reflect updates also on real disk, and to initially load the virtual disk at the begin.

Refer to: How to avoid pitfalls with Virtual Disk

VSE/ESA 1.3 Virtual Disk Exploitation

Virtual Disk for VSAM files under CICS

- **Before using Virtual Disk for such type of files, try to use CICS Data Tables or big VSAM LSR/NSR buffers**

SQL/DS Consideration

- **SQL/DS tables**

SQL/DS allows that specified tables reside on a specific VOLID.

If such a table is READ-ONLY, AND the database setup is such that this table can be separated to reside on a separate disk, one could THEORETICALLY think of placing such a table on Virtual Disk.

This would be a very dangerous situation, since

- if only 1 update is being done

  and

- the virtual disk is gone (Power failure...)

the complete SQL/DS data base may be corrupted (pointer mismatch).

- **Do not place selected SQL/DS tables on Virtual Disk**

  except you are willing and can recreate the COMPLETE data base including all logged updates

- **SQL/DS internal DBSPACEs**

Space for SQL/DS temporary tables can be put on Virtual Disk, such as for performing sorts, produce JOIN composites, and perform view materialization.

They are used for complex queries and maintenance jobs.

VSE/ESA 1.3 Virtual Disk Usage Hints

How to avoid pitfalls with Virtual Disk (VD)

- **Do not grossly overcommit your real storage with VD**

  Impact on VD performance (paging and CPU-time)

  Impact on concurrent CICS online production (incl. VTAM + SQL/DS)

- **Keep in mind that VD essentially avoids physical I/Os**
  - **VD is hardly suited for CICS partitions or for batch partitions with overlapped I/Os**

- **When selecting the type of files for VD:**
  - **Prefer those with the smallest I/O blocking/buffering:**

    Applications using already high buffering are not so suited for VD (8K and more per I/O, and/or many I/O buffers).

  - **Do not use VD to replace efficient VSAM buffering**

    VD access costs I/O supervisor and intercept pathlength
  - **Prefer those programs where I/Os to workfiles are not so overlapped with other I/Os or CPU-time.**
  - **Prefer those which reside on volumes with high contention**

    Benefit also for other partitions if I/O contention reduced
  - **Be careful before moving Sort workfiles to VD, they may be already highly blocked and highly overlapped.**

VSE/ESA 1.3 Virtual Disk Usage Hints ...

How to avoid pitfalls with Virtual Disk (VD) ...cont'd

- **Define your VDs as small as possible**
- **Reuse VD extents as much and as soon as possible**
- **Release a complete VD if data no more required**
  via DVCDN cuu and VDISK UNIT=cuu,BLOCKS=0

  By this you can avoid that, long term, unused data are paged out to the PDS and later unnecessarily paged in again when extent is reused
- **Do not load big virtual disks (e.g. with VSAM B/R ...) during production**
  Loading VDs larger than the unused real storage will cause page-outs and page-ins.

Carefully select Virtual Disk files and setup

Use in any case Virtual Disk for SYSLNK and Label Area

VSE/ESA 1.3 Virtual Disk Usage Hints ...

Reasonable Upper Size for Virtual Disk(s)

Paging aspects

Paging for a virtual disk in general is disadvantageous, since mostly 4K blocking can be achieved with standard access methods and real disks.

Disadvantages exist in any case, if (due to the global LRU page replacement) CICS partitions get page faults, what is very harmful. Overcommittment of real storage with VD will have a negative impact on CICS response times vs CICS paging.

VSE internal load levelling of lower priority batch partitions may reduce this problem but never solve it.

Assessment

- Run production workload without VD
- Determine/Estimate the amount of remaining real storage, start with a total size for all VDs lower than this value.
- Subsequently increase the amount of virtual disk files and monitor overall performance, until optimal.
- Be careful, overcommittment of real storage by VD may impact CICS production suddenly.

  Distinguish between parallel batch by day and night batch.

Cost of Data Space space (if not used)

- VSIZE
- setup of page tables at DSPSERV time
- page management tables in PFIXed private space and real storage
- PDS space on disk

Data in Memory (Summary)

Data In Memory:

Keep more data in storage and save DASD I/Os

CICS Data Tables

Customers with

- intensive full key VSAM KSDS READs in CICS applications

IO reduction: up to 30%	CPUT-reduction: up to 20%

Multiple VSAM LSR Pools

Customers with

- CICS VSAM KSDSs that do not benefit from CICS Data Tables
- need for better tuning of many VSAM files with various request types

IO reduction: up to 30%	CPUT-reduction: up to 10%

Bigger VSAM NSR Buffers

Customers with VSAM NSR files

IO reduction: up to 20%	CPUT-reduction: up to 10%

Reductions very depending on application

VSE/ESA 1.3/1.4 I/O Performance

PART E.

VSE/ESA 1.3/1.4 I/O Performance

VSE/ESA I/O Performance is now discussed in a separate document

'IBM VSE/ESA I/O Subsystem Performance Considerations'
Part of VE21PERF PACKAGE

VSE/ESA 1.3/1.4 Misc. Performance Remarks

PART F.

VSE/ESA 1.3/1.4 Misc. Performance Remarks

Overview

- DEBUG Impact
- FASTTR or NOFASTTR?
- Virtual Disk vs DASD Caching
- More Real Storage or Faster Devices
- POWER Performance
- 'MIPS'?
- 9221 211-based Processors
- Distributed Workstation Feature

DEBUG Impact on Performance

Internal DEBUG Facility for Trouble Shooting

- **Background**

  All VSE supervisors are capable of internally tracing basic events, without using the universal SDAID

```
In case of functional problems, customer may be asked by IBM to
temporarily switch DEBUG on via separately documented AR commands:
e.g.
      - DEBUG             to display DEBUG status (default is OFF)
      - DEBUG ON (,nnnK)  to switch it on
      - DEBUG OFF         to switch it off
      - DEBUG END         to free DEBUG areas in System GETVIS

These commands must work, also when non-IBM products are installed.
If not, immediately contact IBM and the other vendor to resolve.
```

- **Performance Impact by increased CPU-time**

Sample overhead results (DEBUG ON vs OFF)

Base CPU utilization	PACEX1 15%	PACEX4 50%	RAMP-C 80%	DSW 80%
CPU-time /job or tx	+18%	+18%	+12%	+6% e
(variation range)	(5%-25%)	-	-	-
#instr. total	+10%	+11%	+ 5%	+3% e
#instr. in SUPVR stat.	+32%	+34%	+21%	+16%e
Batch ET, tx RT	+ 2%	+ 4%	+12%	+6% e
(variation range)	(0%- 3%)	-	(5%-30%)	-

- Measured results for VSE/ESA 1.3.1 on 9221-150, (e=estimated)
- Values apply to shipped VSE/ESA supervisors. Internal used DEBUG=YES sup's have higher overhead. (SDAID tracing impact is even higher)
- CPU-time overhead depends on 'relative intensity of traced supervisor events'
- Elapsed or Response time overhead depend on CPU-time overhead and CPU-utilization

→ **Overhead in debug situations acceptable (only then)**

To 'Fast' or 'No-Fast Translate'?

Background

- Virtual addresses in channel programs must be translated to real addresses
- The amount of total space for copy blocks (72 byte each) is determined by the BUFSIZE parameter of the IPL SYS command (apart from an upward rounding to a 4K boundary)

  This space is in 'shared space below' and is permanently fixed
- 'Copy-block-bound-conditions' (e.g. seen via STATUS) should be avoided (e.g. increased run-times for batch)

Fast CCW Translation (FASTTR) Performance Functions

- Keeps copies of translated channel programs for possible re-use (within a certain time) by the same task (beneficial if I/O areas are re-used very soon)
- More BUFSIZE may be required for FASTTR to improve 'look-aside', but too many may adversely affect pathlength for CCW translation, especially if no replicas are found (job not suited for FASTTR)
- Any VSE/ESA supervisor has the capability to do FASTTR (until and including VSE/ESA 2.3).
- FASTTR is the system default, except STDOPT FASTTR=NO was set by the system programmer
- JCL Option // OPTION NOFASTTR holds for 1 VSE job (till /&)

Usage of VSE Fast CCW Translation for a job

STDOPT FASTTR=NO	OPTION NOFASTTR	Fast CCW Translation
no	no	YES
no	yes	NO
yes	no/yes	NO

no Option not set/used
yes Option set/used
- Use the QUERY STDOPT command to check STDOPT

To 'Fast' or 'No-Fast Translate'? ...

Recommendations/Conclusions

- **Use FASTTR with care, if at all**
- **Use NOFASTTR for VSAM jobs (batch and CICS)**
  - VSAM (if at all) is expected to benefit the less from FASTTR, the more LSR buffers/subpools are used for DIM. Also, there is no benefit if the location of the channel program varies.
- **Use NOFASTTR for SQL/DS (DB2 for VSE and VM)**

  For SQL/DS VSAM ESDSs, NOFASTTR is beneficial
- **FASTTR only may be beneficial for BAM/SAM batch jobs**

- Sample measurements have shown e.g. that
  - Librarian does not benefit from FASTTR due to its dynamic buffer assignment
  - Performance benefit may be up to about -3% CPU-time only (for an individual batch job, suited for FASTTR)
- Partitions using EXCP REAL are not affected (e.g. POWER)

- **Do not specify huge BUFSIZE values with FASTTR**

  If I/O is not suited for FASTTR, it may hurt even more.
- **Use the SIR command to display the BUFSIZE high water mark**

  Only with STDOPT FASTTR=NO, the true BUFSIZE (=copy block) requirement is shown. Use it after a busy day of production. FASTTR will essentially take all whatever you specify

VSE/ESA 1.3 Virtual Disk vs DASD Caching

How do they compare?

	Virtual Disk	DASD Caching
Areas of usage	All type of 'workfiles' + Reads & Writes ++	All type of files (also permanent) ++ Reads & Writes(3990) ++ Reads (9345) +
I/O duration	0 msec ++	3 - 4 msec for cache hits +
I/O resources required	None, except when paging +	Channel, cached CU, cachable devices
CPU-time deltas (same thruput)	About +- 0% (9221s) Slight increase (9121s)	No delta
Processor storage required	Yes	None +
Application change	None, except updates to real disk +	None ++
Things to watch for	Too high real storage overcommitment will impact CICS paging!! All files of a volume are 'cached' +	Do not cache files not so suited for caching Caching is on (phys.) volume level
Benefit ranking: + = good, ++ = very good		

Conclusions

- Each type of 'caching' has its own area of usage, area of cost, but similar benefits. Both complement each other
  - Use Virtual Disk only where beneficial for 'workfiles'
  - Use DASD Caching for permanent files

> Use CICS Data Tables for full key read intensive VSAM KSDSs
(Exploit the CPU-time and response time benefits vs DASD caching)

More Real Storage or Faster Devices First?

Decision

- **VSE/ESA 1.3 systems benefit from both, complementing each other**

  Both allow to improve Response Times (batch Elapsed Times)
  (or to reduce RT increase when driven to higher throughput)
- **But what to install first?**
  - if budget limit lower than required, and
  - freedom of choice (function-wise)
  - except decision determined by other factors
- **Decision aid (non-exhaustive)**

```
                                 |
                                 V
                        -------------------
 --------        Y     | High CICS paging  |         --------
|        |<-----------| expected ?        |        |        |
|        |             -------------------         | Faster |
|  More  |                      | N                |(or     |
|        |                      V                  | cached)|
|  Real  |              -------------------   N    |        |
|        |             | Workload I/O in-  |----> * | DASDs  |
| Storage|             | tensive ?         |        |        |
|        |              -------------------         | first  |
|  first |                      | Y                 |        |
|        |                      V                   |        |
|        |       Y      -------------------         |        |
|        |<-----------| Very important to |        |        |
|        |             | save CPU-time?    |        |        |
|        |              -------------------         |        |
|        |                      | N (not so much)   |        |
|        |                      V                   |        |
|        |              -------------------   Y     |        |
|        |             | Is WRITE critical?|--------->|        |
|        |             |                   |        |        |
|        |              -------------------         |        |
|        |                      | N                 |        |
|        |                      V                   |        |
|        |   Y (VSAM)   -------------------   N     |        |
|        |<-----------| Does S/W easily   |--------->|        |
 --------              | allow DIM?        |         --------
    |                   -------------------              |
    V                                                    V
 --------                                            --------
| Faster |                                          | More   |
| DASDs  |                                          | Real   |
| later  |                                          | later  |
 --------         * seldom, decision not so critical --------
```


Some Partition and POWER Related Results

- **Dynamic partition startup overhead (vs static)**

  Small increase vs VSE/ESA 1.2, but still only about 1 sec (9221-150)

  No overhead when partition active

- **POWER queue file in partition GETVIS (vs VIO)**

  CPU-time savings of up to 15% for POWER queue commands

- **POWER 4K DBLK size (vs 2K)**

  Only small (1%) overall I/O savings for average batch loads.
  Higher savings for spool intensive jobs,
  and for display of POWER output under CICS or IUI.

- **POWER dispatching trace impact**

  Trace now is started by default (better problem analysis).
  Costs 0.5% CPU-time for average batch jobs.

Improving Spool Performance of SYSLST (3800)

3800 Printing Subsystem

IBM Program 5747-CC1

(Was DOS/VSE Printing Subsystem, 3800 ICR)

Blocks the spooling requests in VSE/AF, before POWER invoked

3800 IOCS modules adapted and standard in VSE/ESA 1.3

All modules are standard in VSE/ESA 2.1.1

Performance benefits

Saves CPU-time to handle POWER spooled lines, also applicable if real printer is not a 3800

Spool intensiveness	3800 vs PRT1 CPU-time	3800 vs PRT1 Elapsed time (9221-150)
low	-3%	+-0%
extremely high	-54%	-28%

Base is VSE/ESA 1.3 with default POWER setup (DBLK=4K, ADD FEE=PRT1)

FCBs

For each form control buffer 2 FCBs have to be generated:

- 1 via IEBIMAGE (5745-SC-IMP, part of 5747-CC1) for spooling
- 1 via Assembler for printing on PRT1

To avoid POWER console messages, be sure to have incorporated DY43856 (PTF UD4986 for VSE/ESA 1.3/1.4). This PTF causes that only at data loss messages 1Q4KI/1Q4LI are displayed.

Refer also e.g. to:

- 'Boosting VSE Performance' by T.Pylant, Enterprise Systems Journal 10/93, pp 74-77
- VSE/ESA 2.1/2.2 Performance Considerations (POWER part)

To 'MIPS' or not to 'MIPS'?

For the more interested reader it may be of interest how many millions of instructions are being processed per second on a processor.

MIPS = IER = Instruction Execution Rate

This, though impressive, is in general a dangerous and dependent item if not used properly, since

- **it heavily depends on the architecture**
  If an instruction set is less powerful, more instructions are needed to complete the same work, and thus higher IER is not a reasonable measure for processor speed.
- **it heavily depends on the job mix or instruction mix**
  For commercial programs many I/O instructions in general are included, thus slowing down the IER of the processor. If only short instructions are executed, the IER is higher than with more complex or heavy instructions. Commercial and engineering/scientific may differ significantly in their IER rate.
- **it also depends on the processor utilization**
- **it does not reflect the speed benefits gained by replacing simple and recurring instruction sequences by so-called microcode-assists.**
  (if you use them you get lower CPU-time, but at the same time also lower IER!)

> Any use of an IER is only useful thus within the same architecture and for well specified environments and instruction mixes.

Only then, there is a certain chance that IER ratios coincide with (reciprocal) CPU-time values and thus processor speeds and capacities.

The only comparable measure of (internal) processor power is the total CPU-time needed for a certain job/task/transaction.

VSE/ESA 1.3 Distributed Workstation Feature

Distributed Workstation Feature (DWF)

- **DWF Rel. 1.0, optional product since VSE/ESA 1.3.2**
- **DWF Rel. 1.0 refresh available with VSE/ESA 1.3.4 and as PTF UN90126 (full product replacement)**
- **Functions**

  Access to POWER queues and VSE libraries

  Execution of POWER and LIBR commands (C/S application)

  Submission and control of VSE jobs

  All actions done in OS/2 workstation environment

- **Performance Enhancements**

  All new graphical interface (GUI)

  → **Improvements of response times by factors**

  Selectable and configurable viewing filters at window initialization

  → **Marked improvement in response time**

VSE/ESA 1.4 Assembly Problem for CICS Tables

VSE/ESA 1.4 Assembly Problem for CICS Tables

- **Problem**
  - **Generating CICS tables (SNT, TCT, ...) may need much higher elapsed time than before**
- **Reason**
  - **Old DOS Assembler internal buffer is too small for CICS 2.3 macros**
  - **Many additional I/Os are required to Assembler work files**
- **Circumventions**
  - **Avoid that a macro loop crosses the boundary of the internal buffer**

    By adding a dummy macro (e.g. MAPCOMR) before DFHSNT TYPE=INITIAL
  - **Use a VSE Virtual Disk for Assembler workfiles**
- **Additional Info**
  - **Problem does NOT show up in other cases**

    VSE/ESA 1.3 with CICS 2.2
    VSE/ESA 1.4 with High Level Assembler
    VSE/ESA 2.1 with CICS 2.3

More info is contained in RETAIN/ASKQ

VSE/ESA 1.3 Release Transition Hints

PART G.

VSE/ESA 1.3 Release Transition Hints

Overview

General Transition Rules

General Rules

Refer also to the chapter 'VSE/ESA Migration Aspects' in the document 'VSE/ESA 1.1/1.2 Performance Considerations'

- **Have relevant measurement data available for your source system (pre-VSE/ESA 1.3)**
- **Do not change w/o need too many environment parameters at the same time**
  (H/W, S/W, I/O setup, buffers, program options....)
  (But note: tuning may be required to have performance-wise comparable/better results, e.g. LSR setup)
- **Adapt your H/W resources (processor power, real storage) to exploit**
  - more concurrently active partitions/tasks
  - Data in Memory (DIM) exploitation
  - Increase exploitation in a controlled manner in order to get most benefit for your environment
- **Do not expect benefits from tuning actions, if you have no bottleneck in that area**

General Transition Rules ...

General Rules (VM/VSE guest only)

- **Do not always consolidate several VSE guests to a single VSE/ESA guest**
  - More than 1 VSE guest may be beneficial on dyadic processors
  - Keep day batch production separate if time critical jobs exist and CPU utilization high (more flexibility by VM dispatching)
- **Try to run VSE production with V=R/F and dedicated I/O devices**
  to get best guest/native ITR ratio.
  With V=V, dedicated devices only bring smaller benefit, if at all.
- **Do not run with V=V without VM CCW Fast Path (CFP)**
  Refer to the VM/VSE Only section!
- **Be aware that MODE=VMESA only offers a single address space**
  - GETVIS of highest partition must start at least 48K below the 16 MB line (partition SIZE at least 80K)
  - VSE CCW translation required for MODE=VMESA and VSE/ESA 1.3
  - MODE=ESA supervisor pre-dominantly required.

General Transition Rules ...

Consolidation Effects of VSE guests

- **Consolidation of several VSE guests into a single guest:**
  - Roughly the same total CPU-time at same total throughput
  - Savings in real storage through single components
    (Supervisor, POWER, VTAM, CICS(s)...)
  - Some CPU-time savings only when
    - 'lower' VSAM share options
    - savings in lock file activity
    - better VM/VSE guest ratios achievable via dedicated devices for preferred guest(s)
    - less VSE/CICS MRO transaction routing or function shipping required (CICS consolidation)

More Specific Migration Rules

More Specific Migration Rules

- **Start with the automatic move of VSAM buffers for 24-bit transactions above the line (VSCR)**
  Only 1 statement (ALLOC) has to be changed
  Observe the results for relief
- **Continue with exploitation of Data In Memory (DIM)**
  - Start with CICS VSAM Multiple LSR pools
  - Selectively use CICS Data Tables
  - Exploit residual real storage with Batch

  Use the DIM items as recommended
  - Increased VSAM buffers above the line
  - CICS data tables
  - Virtual disks (prefer to optimize them AFTER CICS optimization)
  - Other usage of data spaces
- **Relink/Recompile your COBOL II applications for 31-bit**
- Check your Assembler programs for 31-bit conformance before compiling for 31-bit with HL Assembler
- **Check again the opportunities for ESA exploitation**
  (in this doc)

More Specific Migration Rules ...

VSE/ESA 1.3 Vendor Products

- Vendor products are valuable and mostly indispensable additions to VSE in a complete customer production environment
- In spite of close and early cooperation with vendors, in spite of all efforts for quality and performance ...

  individual vendor products may not be available in the desired/required status
- In case of performance problems:

  Try to localize the program product in question:

  - Disable or even de-install (if possible) a product to roughly localize the performance degradation origin

  Check whether the program products are installed correctly

  Example:
  Were the recommended/prescribed phases put into the SVA?
  VSE JCL processing may be slowed down up to 10 times,
  if you do not put JCL intercept phases into SVA, such as
  CADCJXTA from CA DYNAM-T.

  Contact the vendor:

  - Ask for performance information
  - Ask for performance PTFs
  - Request for better documentation (if required)

(Naturally this also applies to IBM)

VSE/ESA 1.3 Release Transition Steps

Possible VSE/ESA 1.3 Transition Steps

(Here CICS oriented, w/o consolidation steps, w/o growth through more terminals/applications)

- **Stage 1**

  Transparent Use of 31-bit for 24-bit applications — VSCR

  - Move VSAM buffers above the line
  - CICS areas above the line

  EFFORT: None, except partition allocation

  BENEFIT: VSCR, about the size of current VSAM buffers
  First relief, with current applications

- **Stage 2**

  Exploiting Data In Memory for 24-bit applications — DIM

  - Provide bigger data buffers
    (Data Tables, Multiple LSR pools, Bigger NSR buffers)

  EFFORT: Some, careful buffer extension,
  check for enough real storage,
  monitor progress

  BENEFIT: Reduced number of I/Os per transaction,
  ITR improvement, better response times

- **Stage 3**

  Exploiting 31-bit Applications (COBOL II, HL-Assembler) — VSCR+DIM

  - Move applications above the line

  EFFORT: Yes

  BENEFIT: Even more VSCR below the line

VSE/ESA 1.3 ESA Exploitation Checklist

VSE/ESA 1.3 VSCR and DIM Opportunities

	Residence	I/O savings for	CPU-time savings for
Bigger CICS DSA below 16M	AS	-	CICS program compr. and SOS processing
Related terminals/ applications/files in same CICS	AS	-	CICS transaction routing and function shipping
More/bigger VSAM buffers (NSR, LSR) for permanent user data on DASD	AS	READs	Setup of I/O, I/O interference, LSR search, if shorter
CICS Data Tables (VSAM LSR KSDS)	AS	READs (full key)	Setup of I/O, I/O interference, VSAM code
More/bigger VTAM buffers in private space	AS	-	Buffer expansions
Virtual Disk for temporary/work data ---> see separate list	DS	READs WRITEs	I/O interference
Vendor 'applications' or enabling software, using data spaces	DS	READs WRITEs	I/O interference
System internal data	AS/DS/ SVA-31	READs WRITEs(temp)	Setup of I/O, I/O interference
31-bit languages	AS	-	VSCR ...
More applications resident	AS	READs	Setup of I/O, I/O interference, Program Load code
Code applications more generous regarding virtual storage	AS	READs WRITEs(temp)	Setup of I/O, I/O interference, application code

AS=Address Space DS=Data Space SVA-31=31-bit SVA

Apart from the last item, all opportunities do not require any change in the application itself.

Rules for Number and Setup of VSE Partitions

Recommendations

- Use any number of CICS partitions best suitable for you

  Consolidate ...
  to get 'lower' VSAM share options, less DASD sharing, less MRO, less storage for CICS base modules...

  Be aware that with 24 bit applications only,
  CICS capacity still is limited

  Keep separate ...
  if wanted/required for handling or setup reasons

  Consider that a single huge CICS partition may not be so easily split in order to run 2 VSE guests under VM on a dyadic processor.

- Batch running concurrently with CICS production:

  Increase #concurrent batch partitions
  only as long as CICS is not affected

  Use all means to separate batch and CICS:
  Setting of PRTY, PRTYIO, file share option and location...

  Batch throughput problems may occur if Online utilization high
  (refer to Workload Balancing part)

- Batch only (e.g. night window):

  Refer to next chart

Night Batch Window

Means to Reduce Required Batch Window

Use 1 or more of the following actions to exploit ESA:

- **Reduce the number of batch I/Os by DIM**
  - More buffers and/or better defined I/O buffer setups (mostly VSAM)
  - Use Virtual Disk wherever appropriate and possible
  - Put Label Area on VSE Virtual Disk
    For VSE/ESA 2.1 use the Label Area in native data space
  - Put VSE Lock File on VM Virtual Disk (VM/ESA 1.2.1)
  - Use VM Minidisk Caching if Virtual Disk functionwise not eligible (VM/ESA 1.2.2)
  - Use bigger POWER DBLK sizes to reduce data file I/Os
    VSE/ESA 2.1 allows DBLK size up to track-size
  - Use VSAM Data Compression (VSE/ESA 2.1)
    if batch processing is at least partly sequential (and CPU power left in batch window)
- **Improve I/O speed**
  - Check current I/O times
    Refer to 'VSE I/O Performance' of this document
  - Use faster DASDs or cached DASDs
  - Use VSAM B/R with S/W data compression
    If CPU-power left and backup jobs are tape-bound

Continued ...

Night Batch Window ...

Means to Reduce Required Batch Window (cont'd)

- **Reduce CPU-time for batch**
  - Use 3800 ICR to reduce POWER spool overhead
    Refer to chart in part 'Miscell. Performance Remarks'
- **Run more batch partitions concurrently**
  - Analyze whether inter-dependencies between jobs still exist or can be reduced e.g. by splitting jobs.
    May require new job-setup (deviation from original scheme) or even a new batch application design
  - Increase #partitions only until CPU full and paging not too extreme (normally no problem for batch-only)
- **Install vendor S/W for automated job scheduling**
  Avoid manual intervention between batch jobs
- **Check tape usage and/or eliminate redundant backups**
  Investigate backup of smaller files to 3390-9s
- **Run some batch already by day (if possible)**
  Careful setup and balancing required, refer to 'balancing charts'

ITR Deltas for Migration

ITR Deltas for Migration

Items to be observed:

- **ITR degradations for individual VSE releases** (caused by increased functionality) **will add up across several VSE releases to higher values**

  These deltas have all been measured for individual workloads and releases, available to your IBM representative.

  They are e.g. also incorporated into the VSE/ESA Quick Migration Sizer, which can be used by your IBM representative to assess such migrations.

- **Moving away from single address space to multiple address spaces (VAE) costs also CPU-time.**

  Especially true under VM, if CP overhead increases:
  when moving
  from MODE=VM (or VMESA), V=V
  to MODE=ESA with V=V, or V=R w/o dedicated devices.

  - **Be aware of the steps you do all at once:**
    - Skipping several major VSE releases
    - Introducing several address spaces
    - Moving to ESA-mode (often without additional tuning)
    - Moving away from a 'full handshake supervisor'
  - **Migrate with appropriately set expectations**

CPU Utilization Aspects

'CPU Utilization'

- **Judge on CPU utilization only related to throughput**
  - **Higher CPU utilization at same throughput: not so good, tuning required**
    Slight increase of CPU-time per tx possible through new functions
  - **Higher CPU utilization at higher throughput: will be seen often ('latent demand')**
    (If you have removed potential newly emerging bottlenecks)
  - **Increasing throughput to a higher degree than CPU utilization:**

    less CPU-time for the same amount of work

    In VSE/ESA 1.3 this can be achieved by
    - Optimal setup and usage of CICS data tables
    - Reduction of lock situations via lockmanager/lockfile
      - reduced lockfile accesses by combining VSEs
      - reduced locking by 'lower' VSAM share options
    - Reduced usage of CICS MRO transaction routing and function shipping
    - Fewer CICS program compressions and short-on-storage cases

Real Storage Usage Notes

Real Storage for VSE/ESA 1.3

- **Notes**

  - **The effective use of more virtual storage requires more real storage**

    Customer benefits with DIM do not come for free

  - **If you do not provide adequate real storage**
    compared to your actively used virtual storage...
    paging will
    - first compensate the DIM benefits

    Paging may result which -in general- uses 'only' 4K blocking. For a batch only system, you still may be better with DIM, even when you have paging (if you otherwise would have blocking (KB/IO) less than 4K).

    - then adversely affect CICS production,
    even if you save some file I/Os via DIM

    Increased paging will also cause some page-outs of CICS pages
    - in spite of 'least-recently-used' paging algorithm
    - in spite of preferred page fault handling for higher priority partitions.

    Such page faults in CICS are much more harmful, since in most cases the whole CICS partition has to stop.

  - **Plan your usage of DIM, especially Virtual Disk**

Recommended Real Storage Sizes

- **The effective use of more virtual requires more real storage**
- **If you do not provide adequate real storage, paging will**
  - first compensate the DIM benefits
  - then adversely affect CICS production

- **To still effectively exploit 'some DIM' use about twice the real storage as compared to /370**

Processor	Recommended Real Storage (Native VSE/ESA)
(4381-91E)	(64 MB)
9221-120	32 MB
-130	32 MB
-150	64 MB
-170	64 MB
-191	128 MB
-201	128 MB
-211	128 MB
9672-R11	128 MB
9672-RA2	128 MB
9672-R12	128/256MB
9121-180	64 MB
-190	64 MB
-210	128 MB
-260	128 MB
-320	128 MB
9121-311	128 MB
-411	128 MB
-511	256 MB
9021-711	256/512MB

Higher values still may be beneficial (More DIM, under VM)

VSE/ESA Paging Performance

- **Best paging performance is a low page-I/O rate**

  'Best paging is no paging'

  > Provide sufficient processor storage

- **Monitor your page-I/O rate**
  e.g. with the Display System Activity Panel
  with any performance monitor

- **Still acceptable paging:**

  up to 6 page-I/Os per (independent) CICS partition

  Details contained in the VSE/ESA 1.1/1.2 Performance Considerations,

- **Page Data Set (PDS) layout**

  consider the PDS layout of the Base Install only as starting point

  use as many PDS extents as possible, each on a separate device

  define PDS extents with equal sizes

  select volumes with low file-I/O activity

  caching of PDS extents may not be very effective regarding hit ratios (except when thrashing with high rate)

- For further info refer e.g. to Enterprise Systems Journal 01/93

VSE/ESA 1.3 Performance APARs/PTFs

Some APARs/PTFs of performance interest

```
* PN35557     UN38140     CICS GETVIS above the 16M line
              UN38141

   This PTF causes CICS/VSE 2.2 to use several GETVIS subpools
   and thus to avoid long (CPU-time intensive) GETVIS searches
   for frequent and small GETVIS requests above the line.

* PN37166     UN42053     CICS Short on Storage
              UN45695     (Standard in 1.3.2)

   This PTF fixes a problem where TS table entries were not freed and
   thus caused SOS (DSA shared subpool)


The following items are part of 'PTF1A' (UD48317, UD48312):

* DY42372     'PTF1A'    VSE load levelling

   Under specific situations, a partition could be deactivated too
   early.

* DY42372     'PTF1A'    ICCF high priority task

   This PTF makes sure that the ICCF high priority task is properly
   dispatched also at ICCF low activity periods.
   Avoids erratic long elapsed times in a single ICCF I/A partition.


The following items are part of 'PTF2' (UD48399, UD48402),
VSE/ESA 1.3.1:

* DY42426     UD48403    INFO/ANA Dump viewing

   Gives faster response time to list dump entries.

* DY42390     'PTF2'     Label Area on Virtual Disk

   JCL extensions to allow label area on Virtual Disk

* DY42402     'PTF2'     Lock file on ECKD DASD

   This PTF provides native ECKD channel programs for accesses to
   the lock file, and thus avoids lost revolutions for WRITEs
   to the lock file on ECKD
```

VSE/ESA 1.3 Performance APARs/PTFs ...

Some APARs/PTFs of performance interest (cont'd)

* DY42602 UD48658 Virtual Disk with large blocksize

  This PTF saves some CPU-time when bigger blocks are written to a virtual disk.

* PN43369 UN50149 Missing disconnect in CICS for POWER SAS

  This PTF assures that the CICS spooler always disconnects from the POWER Spool Access Support. It may have caused POWER private storage problems and longer POWER task searches.

* DY42982 UD48882/3 High number of VSAM index-I/Os

  Under certain conditions (BUFSP or BUFNI not explicitly specified and number of index levels increased) too many index I/Os for KSDS and VRDS could have occurred.

The next 2 PTFs became available 03/94 and refer to DASD caching with VSAM (thus the PTFs are standard since VSE/ESA 1.3.5):

* DY43072 UD90363 VSAM support for 3990-6 Enhancements

  This PTF provides the VSAM support for 'regular data format' and for 'record cache mode' of the 3990-6 enhancements. Also, seq. bits are set for better cache control during VSAM SPEED load mode. This PTF installed (or by default included since 1.3.5) requires a 9340 u-code patch (E6392AC)

* DY43138 UD49025 VSAM B/R cache bit settings for ECKD

  This PTF uses the sequential caching bits instead of bypass cache in order to speed up Backup(!) to a target disk. It also applies to 9345 Cache, which in its latest EC 486392 adequately exploits the sequential setting.

The next PTF was closed 03/29/94 and thus is not included in VSE/ESA 1.3.4. It refers to several specific problems.

* DY42963 UD49049 to UD49052 Miscellaneous fixes

  This PTF avoids that in connection with block paging, under certain specific conditions, 'empty' page-ins were done. Also a System GETVIS problem was removed, occurring with FAQS PCS. See DY43137 for complementary PTFs.

VSE/ESA 1.3 Performance APARs/PTFs ...

Some APARs/PTFs of performance interest (cont'd)

The next PTF was closed 11/14/94 and is contained in 1.3.6:

* DY43312 UD49234 UD49237 PTFs retrofitted from VSE/ESA 2.1

  This PTF contains also an enhancement of the CKD/ECKD conversion routine, beneficial for WRITEs with specific CKD channel programs (e.g. CICS journal)

The next PTF was closed 02/10/95 and is not included in 1.3.6.

* DY43414 UD49333 VSAM B/R restore performance for 3990-3

  This PTF sets the beginning of the extent address in the DEFINE EXTENT CCW for VSAM B/R to the begin of the current extent, in order to allow an optimal sequential de-staging for 3990-3 compatible cached control units during RESTORE (of specific importance for RAMAC Array DASD)

* DY43335 UD49325 UD49332 RAMAC Array DASD and Format Writes

  This PTF corrects a problem in the RAMAC Array DASD, which loses a revolution when a standard R0-record is written and a specific bit is not set. Make sure that VM APAR VM60996 is installed for VM MDC.

* DY42800 UD48965 UD48966 VSAM Load mode performance for CI-mode files

  This PTF allows that multiple CIs are chained in the same VSAM channel program when loading or pre-formatting a CNV opened file. It was retrofitted from VSE/ESA 2.1 to VSE/ESA 1.3

* DY43207 UD49163 UD49164 IPL accepts ADD cuu,ECKD for 3380 devices if attached to an ECKD capable synchronous control unit.

  Further functional enhancements are included in this fix.

* PN61694 UN73757 Correction of CICS file statistics
* PN67405 UN68618(PE) after overflow

  UN73757 corrects a PTF which extends file statistics counters beyond 99,999.

VSE/ESA 1.3 Performance APARs/PTFs ...

Some APARs/PTFs of performance interest (cont'd)

* DY43555 UD49523..26 Misc. PTFs, including a GETVIS improvement

  This PTF includes besides other patches a pathlength improvement for GETVIS, applicable for general GETVIS requests (-24, -31)

* DY43836 UD49763 VSAM I/O performance for ECKD format writes

  This PTF corrects a VSAM sector value when doing format WRITEs to ECKD attached devices. It applies to all ECKD DASD attachments and especially to RAMAC Array Subsystem. VSAM REPRO is affected and formatting of new extents, no impact for VSAM B/R Restore

* DY43416 UD49348 VSAM performance improvement for CNV load mode

  This PTF allows chaining of several CIs when loading a VSAM file with MACRF=CNV (CI-processing) and VSAM buffering (MACRF=NUB).

  It applies especially to ADSM/VSE if more disk space is acquired via DEFINE VOLUME.

* DY44070 UD49933 VSAM catalog mgmnt, VSAM managed file on ECKD

  This PTF corrects some VSAM catalog management problems and provides channel program enhancements for VSAM managed SAM files on ECKD devices (sector value for RPS).

* DY44358 UD50212/215 Misc patches, including SIR RESET

  This PTF allows to SIR RESET dynamic counters

This list is provided to give fast hints to resolved performance problems. PTF numbers may have changed, so always refer to APARs when ordering fixes.

Avoiding Pitfalls in VSE/ESA 1.3 and up

How to optimally exploit VSE/ESA 1.3 and up

- **Run VSE/ESA in ESA-mode**

  /370-mode is constrained (16 MB real and virtual ...)

- **Run VSE on ES/9000 (optimal ESA implementation)**
- **Use sufficient real storage for DIM**
- **Select appropriate processor power to allow increased throughput**
- **Exploit 31-bit addressing for more VSCR and for DIM**
- **Use VSE Virtual Disk as recommended**
- **Use as many CICS partitions as required/wanted**
- **Avoid CICS paging**
- **Do not combine critical day batch with high CICS load in a single VSE without enough processor power**
- **Carefully setup partition priorities via PRTY**
  - By new setup or use of DIM workload balancing may have to be re-adjusted
- **Use a VSE system performance monitor (EXPLORE/VSE by Legent, TMON/VSE by Landmark, or equivalent)**

VM/VSE Only Considerations

PART H.

VM/VSE Only Considerations

These charts have been moved into a separate document

'IBM VSE/ESA VM Guest Performance Considerations'

which remained in this VE13PERF package on IBMVSE tools disk.
It is also available as a separate file VEVMPERF.ZIP in INTERNET.

Overview

- Performance value-add of VM
- Guest setup
- VM CCW Translation Bypass & I/O Passthru
- VM 'normal' and 'fast' CCW Translation
- CCW Fast Path and ECKD
- 'Double Paging' for V=V guests
- Reasons V=V is used
- Virtual Disks for VM/VSE
- VM/ESA 1.2.2 Fulltrack Minidisk Caching

VSE/ESA Workload Balancing

PART I.

VSE/ESA Workload Balancing

Overview

- What is a balancing problem?
- Potential reasons for impact

- VSE processor dispatching priorities
- VSE I/O dispatching priorities
- VSE 'System Balancing'

- Principal balancing means
- Which means in which cases?
- 9345 DASD Cache balancing aspects

VSE/ESA Workload Balancing

What is a balancing performance problem?

A special tuning problem, where

- NOT the total throughput of a system is the problem
- NOT the throughput/performance of individual loads is the problem
- BUT the individual throughput/performance of partial loads, when run concurrently

„ **2 types of activities (loads)**

'base load'
(e.g. normal online production)

'add-on load'
(e.g. batch or long running, often massive sequential transactions)

„ **Symptoms**

Each load alone performs good
e.g. it requires not more CPU-time and I/Os than before or expected

If run concurrently, base load is degraded too much or more than before.
Slight degradation of the base load is to be expected in any case, but for some reason the add-on load is too fast and thus the degradation on the base load is too high.

A degradation of the add-on load is not only tolerated, but often desired

A certain throughput is required in parallel to the base load, though.

VSE/ESA Workload Balancing ...

Potential reasons that a balancing problem newly arises

„ **NOTE: Make sure that ...**

the 'balancing' problem is NOT a standard 'resource consumption' problem

i.e. NOT the individual load is already a problem.

E.g. not a 'CPU-time' problem, where for whatsoever reason a component or function requires much more CPU-time to do the same work
(e.g. new release, new application, new setup, missing PTF ...).

„ **More VSE partitions consolidated or set up in a single or several VSEs**

„ **Higher priority load much more CPU-time bound (better: less I/O bound) than before by using DIM, e.g. via Virtual Disk**

„ **I/O device speed has changed**
Slower to faster devices, non-cached to cached devices

í **In all cases (re-)check priority settings**
Start with CPU priorities
PRTY within VSE, SET PRIOR under VM

„ **ECKD DASDs are used newly under VM/ESA**
Make sure the VM/ESA fix VM57265 for CCW translation has been applied
(Too high CP overhead, often producing a VSE balancing problem)

Potential Reasons for Impact

Potential Reasons for Impact on Balance

Ù **Processor storage**
The available processor storage is too small to have both loads concurrently active. Such a case shows up with increased paging.

- Apart from reducing I/O buffers, there are only very limited means available to reduce the working-set of workloads.
- If e.g. too much batch causes CICS pages to be paged out, this in general has a significant impact on CICS response times. Naturally the page manager prefers those pages for page-out which have not been referenced for a long time, but still less often used CICS pages might be selected.
- If paging is very high, VSE load levelling will deactivate the lowest priority batch partitions.

Ù **Processor capacity**
The system provides more concurrently dispatchable work than currently is dispatchable with regard to CPU utilization (which may approach 100%).

- If dispatching priorities are properly set, the problem never is the fact that the lower priority work exploits all processor power currently not usable by the higher priority base load.
- The problem is (apart from potential locking of common required resources) the fact that the lower priority load handicaps the base load in the I/O subsystem (I/O contention, see below).
- Each change in the I/O part (see below) will impact the dispatchability on the processor side.

  Check whether the base load has become more I/O dependent, such that the impact of the add-on load is higher.

Potential Reasons for Impact ...

Potential Reasons for Impact (cont'd)

Ù **Channel contention**
The add-on load puts additional burden on the same channel(s) as used by the base load and thus creates a channel bottleneck, which may hurt the base load more than the add-on load.

Ù **Device contention**
The add-on load uses files on the same device(s) and creates a device bottleneck.

NOTE: Increasing the resources above always may help, but may not be economic, since e.g. the additional resources may be required only part time.

In the cases below, tuning is more difficult and may even require e.g. a different application/operation setup.

Ù **'File contention'**
The add-on load uses the same files as the base load.

- If by accesses to the same file, a device bottleneck is created, you may distribute the same file across multiple devices.
- If both loads are in different VSE partitions, I/O definitions may be selected differently.
- If both loads are in the same partition, only in specific cases distinction in I/O setup can be done, e.g. if the major types of I/O requests are different (random vs sequential, read vs write...)

Ù **Other contentions**
There may be other reasons for degradation, such as LTA contention, contention via a LOCKed resources etc ... These are specific cases.

NOTE: More than one of the reasons cited above may apply at the same time.

VSE Balancing Basics

VSE Processor Dispatching Priorities

The currently highest priority task is dispatched
Within the single VSE partition balancing group, the priorities are rotated on a time slice base (MSECS). So, equal priority of partitions is achieved in practice.
Note that a partition balanced dynamic class gets only the same long term dispatching priority as a single static partitition, even if several dynamic partitions are active in that class (changed with Turbo Dispatcher).

Any I/O interrupt causes a new dispatch decision

í **No high priority task must wait since a low priority task owns the processor**
(except the low priority task would occupy e.g. the LTA)

A partition can occupy the processor for any interval of time, but only until higher priority work is ready for dispatch.

No lower priority batch occupies the processor as long as higher priority CICS has something to dispatch

No capping, no limitation of usage
If there is partition work to be done by the processor, it is done, except the (batch) partition has been deactivated by the system for reasons of load levelling (paging).

Capping is only done for total VSE machines
(i.e. not dispatching a task in spite of an available processor)
- by VM/ESA 2.2, if specified
- for PR/SM LPARs, if specified

í **The only way to (partly) compensate if a processor is not powerful enough:**

Put critical day batch into a separate VSE
(where dispatching priorities are 'relative' rather than 'absolute')

VSE Balancing Basics ...

VSE I/O Dispatching Priorities

All I/O requests are queued in the VSE 'channel' queue
I/Os are scheduled on a 'rotating PUBSCAN' base
Per device, the highest priority I/O request is selected

I/O priority is highest for system tasks
(SVC15 headqueue priority for page-I/Os, etc)
then normal tasks are selected
Normal task or SVC0 priority within a device is
- according to PRTYIO, if set
- FIFO (first-in first-out), else
Note that the partition priority from PRTY is not used for I/O dispatching.

Highest priority I/O is always started, if the device is idle.
Any started I/O is completed and not interrupted,

if a higher priority I/O for the same device would arise later (This effort would be not justified/create too much overhead.).

í **In spite of PRTYIO, I/O contention can only be reduced, but not completely avoided**

í **If a lower priority load impacts higher priority load, in most cases I/O contention is THE reason**

Therefore,
Reduce I/O contention by
- rearrangement of DASDs, files
- using Data In Memory (VSE/ESA 1.3)
- increasing I/O resources

Principal Balancing Means

The following balancing means are available, though not for each case applicable or of impact.
Please note that under each key-word more than a single tuning activity may be understood here.

1. VM or LPAR priority settings

CPU dispatching priorities can be set

- via SET PRIOR, SET FAVOR, SET SHARE, QUICKDSP ...
- via Favored Partition, Processing Weights, Wait Completion ...

Note if there is still enough processor power available, only in a PR/SM LPAR, VSE can be capped in processor usage.

2. PRTY settings

Set VSE partition dispatching priority. You either may increase base load or decrease add-on load dispatching priority.

3. PRTYIO settings

Introduce different I/O priorities into the channel queue for dispatching I/Os.

4. Usage of TPBAL not recommended

Refer to VSE/ESA 2.1 document

5. PCT priority

In the CICS PCT, it is possible to define different user/terminal/tx priority. Also it may be possible to introduce CICS DELAYs into a massive sequential transaction, if other means do not help enough.

6. Distribution of devices across channels

Separating I/Os from both workloads onto different channels to remove or reduce channel contention.

7. Distribution of files across devices

Avoiding/reducing contention of I/Os at device level.

8. I/O and file definitions

Providing 'better' I/O definitions (number of buffers, strings, CI-size, ...) for the base load or 'worse' I/O definitions to slow down the add-on activities.
It also may include definitions in the ACB/RPL or FCT regarding the type of I/O requests (e.g. to avoid VSAM NSR read ahead).

Principal Balancing Means ...

9. Device response times

This can be influenced e.g. by different usage of a cachable device (e.g. switch caching off/on, ...)

10. Fewer add-on partitions

If the add-on load consists of several partitions and all other balancing means do not help, it is always possible to start or allow fewer add-on partitions (e.g. only 3 instead of 5 concurrent compiles in a dynamic partition class).

Balancing means in case of 'File Contention'

„ **File contention is a specific, nontrivial balancing situation**

If all other means are exhausted (see 'Principal Balancing Means'), only the following balancing/tuning efforts may help, if possible at all

„ **Make sure, that when too many I/Os go to the same VSAM file, the file is optimally split across several devices:**

- put index and data component separate
- split up the file via VSAM KEYRANGEs

„ **Make sure, the file is on an 'appropriate' device**

in order to reduce the impact of the add-on load

In some cases, the appropriate device is a fast device, in other cases it may be a slower device (e.g. uncached).

In any case, it must be a device type where

the add-on I/O load is more degraded (not as fast) as the base I/O load

the base I/O load is as as fast as required if being run alone

„ **Make sure file definitions are 'appropriate' to your balancing targets**

Principal Balancing Means ...

Balancing means in case of 'File Contention' (cont'd)

„ **Example**

Base load I/O is mostly random

Add-on load I/O is mostly sequential

Potential means for differentiation

If the add-on load is in the same partition, the only chance is to find out, whether the performance parameters for the file can be selected such that they are more favorable for the base load than for the add-on load:

i.e. prefer base-load (or handicap add-on load) by

Add-on load in ...	other VSE	other partit.	same partit.
Smaller data CIs (CI-size)	X	X	x
Less data buffers (NSR BUFND)	X	X	x
Move to LSR (1) (NSR -> LSR)	X	X	x
Define Skip Seq. in ACB (2)	X	X	x

(1) Reduced sequential performance with LSR, since no VSAM Read Ahead is done with LSR
(2) Skip Sequential instead of Sequential in ACB. VSAM does not set the Sequential caching bit (applicable for cached ECKD DASDs at 3990-3/6, if such a problem should occur at all)

In case of VSAM SHROPT 4, the file is locked on a CA-base, which may impact the base load.

Which means in which cases

Balancing means

Select the table and look up which class of balancing means may apply, depending on the observed bottleneck(s):

X means 'potentially of primary help'
x means 'indirect help'
\- means 'no help'

„ **Add-on load in other VSE**

	Type of bottleneck/contention			
	Processor	Channel(s)	Device(s)	File
VM/LPAR priorities	X	x	x	x
PRTY settings	-	-	-	-
PRTYIO settings	-	-	-	-
PCT priority	-	-	-	-
Devices at channels	x	X	-	-
Files on devices	x	-	X	-
I/O and file def's	x	x	x	X
Device RTs	x	x	X	-

„ **Add-on load in other partition of same VSE**

	Type of bottleneck/contention			
	Processor	Channel(s)	Device(s)	File
VM/LPAR priorities	-	-	-	-
PRTY settings	X	x	x	x
PRTYIO settings	x	X	X	x
PCT priority	-	-	-	-
Devices at channels	x	X	-	-
Files on devices	x	-	X	-
I/O and file def's	x	x	x	X
Device RTs	x	x	X	-

„ **Add-on load in same partition**

	Type of bottleneck/contention			
	Processor	Channel(s)	Device(s)	File
VM/LPAR priorities	-	-	-	-
PRTY settings	-	-	-	-
PRTYIO settings	-	-	-	-
PCT priority	X(CICS)	x	x	x
Devices at channels	x	X	-	-
Files on devices	x	x	X	-
I/O and file def's	x	x	x	X
Device RTs	x	x	X	-

9345 DASD Cache Balancing Aspects

9345 Caching and Massive GET Sequential Applications

„ **Symptoms**

Much faster response times for massive sequential transactions

Better response times for normal production transactions

BUT: High impact on production when both run together

CICS transactions (e.g.) which issue thousands of file requests (reading big amounts of data sequentially) run much faster than before, since 9345 DASD caching provides high read hit ratios.

When run in parallel to other production, though these sequential transactions are faster than before with the uncached or elder devices, the (shorter) impact on the other production is high.

Due to the high hit ratio, the CPU utilization for such sequential transactions increase, also the physical I/Os to the devices.

„ **Causes**

'Other production' is temporarily slowed down due to ...

- contention for the CPU

Is only relevant if 'other production' does not run in a partition with higher priority.
Especially is relevant if all production runs in the same CICS partition (see separate bullet)

- channel and/or device contention

Sequential work only may require few CPU cycles in order to issue subsequent requests.
Therefore partition priorities often cannot help too much

- cache domination of sequential applications

Sequential bit in DEFINE EXTENT was not correctly used to limit number of sequential tracks in cache.
Random tracks were cast out too early

í **Apply Microcode Fix EC486392**

9345 DASD Cache Balancing Aspects ...

„ **General Note**

This is NOT a problem of CPU-time, it is a problem of balancing. Higher throughput at same total CPU-time always means higher CPU utilization.

This preference effect is higher if for the sequentially used file

- VSAM NSR is used (with seq. Read-ahead)
- more buffers are used
- CI-size is high
- record size is small

„ **Note to CICS task dispatching**

Most of these type of problems in the field have been solved by the 9345 Microcode fix EC486392.
Nevertheless the fast 9345 device response times may leave balancing problems within a single CICS partition:

- CICS GETNEXT requests are completed so fast that when CICS returns to the dispatch queue, the previous tx can be and is re-dispatched (if no other tx has higher priority).

  Since the default priority of a tx is 1 (i.e. lowest), this problem exists if the other production transactions have not explicitly been assigned to a higher priority value.

9345 DASD Cache Balancing Aspects ...

„ **Circumventions**

- prefer production and/or
- handicap the massive sequential transactions

both on CPU and IO-side

Select those means most appropriate to your environment:

Lower terminal/user/transaction priority

Define lower priority for the sequential transactions in the CICS PCT.
This should prefer normal production for CICS dispatching and thus reduce the I/O rate of the sequential transactions.
Since such sequential transactions are often not very CPU-time intensive, this may not help alone.

Better I/O buffering for normal production

This helps to better prefer normal production within the CICS partition.

Isolate files for sequential access, if possible

Determine which files the sequential transactions use and separate them as good as possible from the rest (separate volumes, even if files used concurrently). This can reduce device contention for normal production.

Insert CICS DELAYs in massive sequential transactions

If all does not help (and still the sequential transactions must run during normal production), insert some CICS DELAYs.
Use e.g. 2 sec delay each 10000 records.

PRTY/PRTYIO settings

Use of the PRTY/PRTYIO commands can only help when such sequential applications run in a separate partition (e.g. batch).

9345 DASD Cache Balancing Aspects ...

„ **Circumventions (cont'd)**

Do not cache individual devices

Switch caching off for those devices where the massive sequential files are located. This reserves the total cache size for all other production volumes.

For cached 9345 subsystems, caching for a device can only be switched off by the CE, and is mostly a means for problem determination only. But, in spite of this, the cache storage is still available for all other devices.

Change VSAM buffering definitions

- smaller CI-sizes and/or smaller buffers
- use VSAM LSR where no read ahead is done
- if VSAM NSR must be kept, define FCT SERVREQ also as READ to avoid VSAM read ahead

Provide more available processor power to normal production

- by faster processor
- by better VM/VSE guest ratios (go to V=R with dedicated devices)

Summary

PART J.

Summary

Summary

- **Significant constraint relief and capacity increases for single VSEs and CICS partitions**

  via more private space below the line

- **High potential for improved response times**

  via Data In Memory exploitation

  via support of fast DASDs (3390/9345s/RAMAC) and Dynamic Channel Subsystem

- **Consolidation of several VSEs or CICS partitions possible, though 1 VSE/ESA should not always be the aim**

- **Stepwise migration into 31-bit exploitation possible**

- **All VSE/ESA benefits optimally exploitable on ES/9000**