IBM VSE/ESA V2 Performance Considerations

Wolfgang Kraemer

VSE Product Mgmnt
Dept 3221
71032-15 Boeblingen
WKRAEMER at BOEVM4
wkraemer at de.ibm.com

Tech Conf Update 2000-03-31

RETURN TO INDEX

Contents

A. H/W Support

B. VSCR for Increased Capacity (incl. DL/I)

C. Enhanced Label Area and LTA

D. Misc. Items for VSE Base

E. VSE/POWER

Contents

F. CICS/VSE 2.3

G. ACF/VTAM 4.2

H. Languages and LE

I. Misc. VSE/ESA Products

Contents

J. Appendix A: VSE Space Optimization

K. App. B: VSE and SAP R/2 Performance

L. App. C: VSE/ESA 2.3 Perf. Enhancements

Contents

M. App.D: Parts moved, or in other documents

Note

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

It is intended and required to update the performance information in this document.

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

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

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

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

TOOLS SENDTO BOEVM3 VMTOOLS IBMVSE GET VE21PERF PACKAGE

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

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

or directly via FTP links

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

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

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

'IBM VSE/ESA 1.3/1.4 Performance Considerations'
'IBM VSE/ESA V2 Performance Considerations' (this document)
'IBM VSE/ESA Turbo Dispatcher Performance'
'IBM VSE/ESA I/O Subsystem Performance Considerations'
'IBM VSE/ESA VM Guest Performance Considerations'
'IBM VSE/ESA Hints for Performance Activities'
'IBM VSE/ESA TCP/IP Performance Considerations'
'IBM DFSORT/VSE Performance Considerations'
'IBM VSE/ESA CICS Transaction Server Performance' (new, 12/98)
'IBM VSE/ESA V2.5 Performance Considerations' (to come)

The files are
VE13PERF.PDF, VE21PERF.PDF, VE21TDP.PDF, VEIOPERF.PDF, VEVMPERF.PDF, VEPERACT.PDF, VETCPPER.PDF, VESORTP.PDF, VECICSTS.PDF, VE25PERF.PDF

Disclaimer

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

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

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

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

Acknowledgements

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

Special thanks to Hanns-J. Uhl for the numerous performance measurements done.

All mistakes and inaccuracies in this document are my own.

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 VSE/ESA performance documents

Wolfgang Kraemer, IBM VSE Development, Boeblingen Lab, Germany

Notes ...

Base Documents

This document essentially deals with the performance differences between VSE/ESA 2.1 and VSE/ESA 1.3.

The document establishing the base performance level of VSE/ESA 1.3 versus VSE/ESA 1.1 and 1.2 is

'IBM VSE/ESA 1.3/1.4 Performance Considerations'

And for differences in VSE/ESA 1.1 and 1.2 performance versus VSE/SP, refer to

'IBM VSE/ESA 1.1/1.2 Performance Considerations'

These documents are also available to any IBM person, as part of the VE12PERF/VE13PERF PACKAGEs on the same IBMVSE TOOLS disk. Contact your IBM representative to retrieve a copy for you by entering the following CMS command:

```
TOOLS SENDTO BOEVM3 VMTOOLS IBMVSE GET VExxPERF PACKAGE
```

The latest update of these 2 packages are dated 03/93 and 10/96.

These documents contain references to further VSE performance documents.

Trademarks

The following terms included in this paper are trademarks of IBM:

ES/9000 ESA/390 System/390 SQL/DS PR/SM
VM/ESA VSE/ESA ESCON ECKD RAMAC
CICS

EXPLORE/VSE is a trademark of Legent Corp. / Computer Associates
TMON/VSE is a trademark of Landmark Corporation
R/2 is a trademark of SAP AG, Walldorf, Germany

What has been added/changed?

Deltas as compared to earlier versions

Editorial changes are done throughout the document without special notice

„ Updates as of 99-03-29

DASD Sharing with different VSE releases

POWER part foil arrangement

POWER Shared Spooling and PNET

Added LTA performance charts

„ Updates as of 99-06-25

New LE/VSE performance PTFs

Part on VSE sequential disk files

„ Updates as of 99-11-15

VSE Librarian performance

VSE Dynamic Partitions and Tasks

„ Updates after 99-11-15

Moved some items to the new document on VSE/ESA 2.5

- 'Dynamic Partitions and Tasks'
- Seq. Disk part
- Librarian part

VSE/ESA 2.4 still in CICS TS document

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

NVS Non Volatile Storage

PR/SM Processor Resource Systems Manager
An ES/9000 standard feature for logical partitioning

RAID Redundant Array of Independent/Inexpensive Disks

RAMAC RAID Architecture with Multilevel Adaptive Cache

VSCR Virtual Storage Constraint Relief
All that provides effectively more space below the 16 MB line, or reduces the space requirements there

Items NOT contained in this document

The following VSE/ESA performance aspects are NOT contained in this document.

- ## VSE/ESA 31-bit Exploitation for VSCR and DIM

  In VSE/ESA 1.3/1.4 document

- ## VSE/VSAM detailed tuning hints

  Refer to Dan Janda tuning documents and to 'VSE/VSAM User's Guide and Application Programming' SC33-6732-00, Chapter 6

- ## VSE/ESA I/O Performance

  In a separate document

- ## VM/VSE Only Considerations

  In a separate document

- ## VSE/ESA Workload Balancing

  In the VSE/ESA 1.3/1.4 document

- ## CICS/VSE Component Performance

  In CICS/VSE manuals, including 'Performance Guide'

- ## VSE/ESA V2 Turbo Dispatcher

  In a separate document

- ## VSE/ESA Perf. Trouble Shooting, Monitoring, Sizing

  In a separate document

- ## TCP/IP Performance with VSE/ESA

  In a new separate document

- ## DFSORT/VSE Performance

  In a new separate document

- ## CICS Transaction Server Performance

  In a new separate document

Further References

The following are references for further performance information in the context of VSE/ESA V2.

General VSE/ESA 1.3/1.4 performance documents are listed in the VSE/ESA 1.3/1.4 Performance Considerations document.

- VSE/ESA V2 Presentation Guide
  VSEG PACKAGE on MKTTOOLS disk

- A Technical Guide to ESA/390 Compression
  GG24-4130-00, Red Book, ITSO Poughkeepsie, 04/94, 120 pages

- VSE/ESA Performance Potpourri
  GUIDE Reno, 03/95 Session VS501, by Dan Janda, 40 pages

- CICS/VSE Multiple LSR Buffer Tuning
  GUIDE Reno, 03/95 Session VS401, by Dan Janda, 40 pages

- VSAM Performance Tuning for the Experienced VSAM Tuner
  VSE/ESA Techn. Conf. 05/95 Atlanta, by Dan Janda, __ pages

- VSE/VSAM Performance and Tuning, by Dan Janda,
  WAVV Conference 10/96, Green Bay, Wisconsin, 70 pages
  VM/VSE Tech Conf 05/97, Kansas City, Sessions 33F-G
  VM/VSE Tech Conf 06/97, Mainz, Germany, Sessions 53F-G
  WAVV Conference 11/97, Chattanooga, Tennessee

- VSE/VSAM Fundamentals and Tuning, by Dan Janda,
  WAVV Conference 10/98, Albany, NY

- VSE/VSAM and CICS/VSAM Performance: Using Shutdown Statistics
  VM/VSE Techn. Conf. 10/96 Rome, by Horst Sinram and Dan Janda, 65 pages

- Data Compression in a VSE/ESA 2.1 Environment
  VSE/ESA Techn. Conf. 05/95 Atlanta, by Dan Janda

- VSE/VSAM Concepts and Tuning Tips,
  WAVV Conference 10/99, Cincinnatti, by Dan Janda, 110 pages

References ...

- 31-bit Addressing in PL/I for VSE
  GG24-4271-00, ITSO Boeblingen Red Book, 01/95, 83 pages

- IBM Cobol for VSE/ESA
  VSE/ESA Techn. Conf. 05/95 Atlanta, by Jim Alexander et al, 33 pages

- High Level Assembler
  VSE/ESA Techn. Conf. 05/95 Atlanta, by Jim Alexander et al, 25 pages

- PL/I VSE
  VSE/ESA Tech Conf 05/95, Atlanta, by Jim Alexander et al, 31 pages

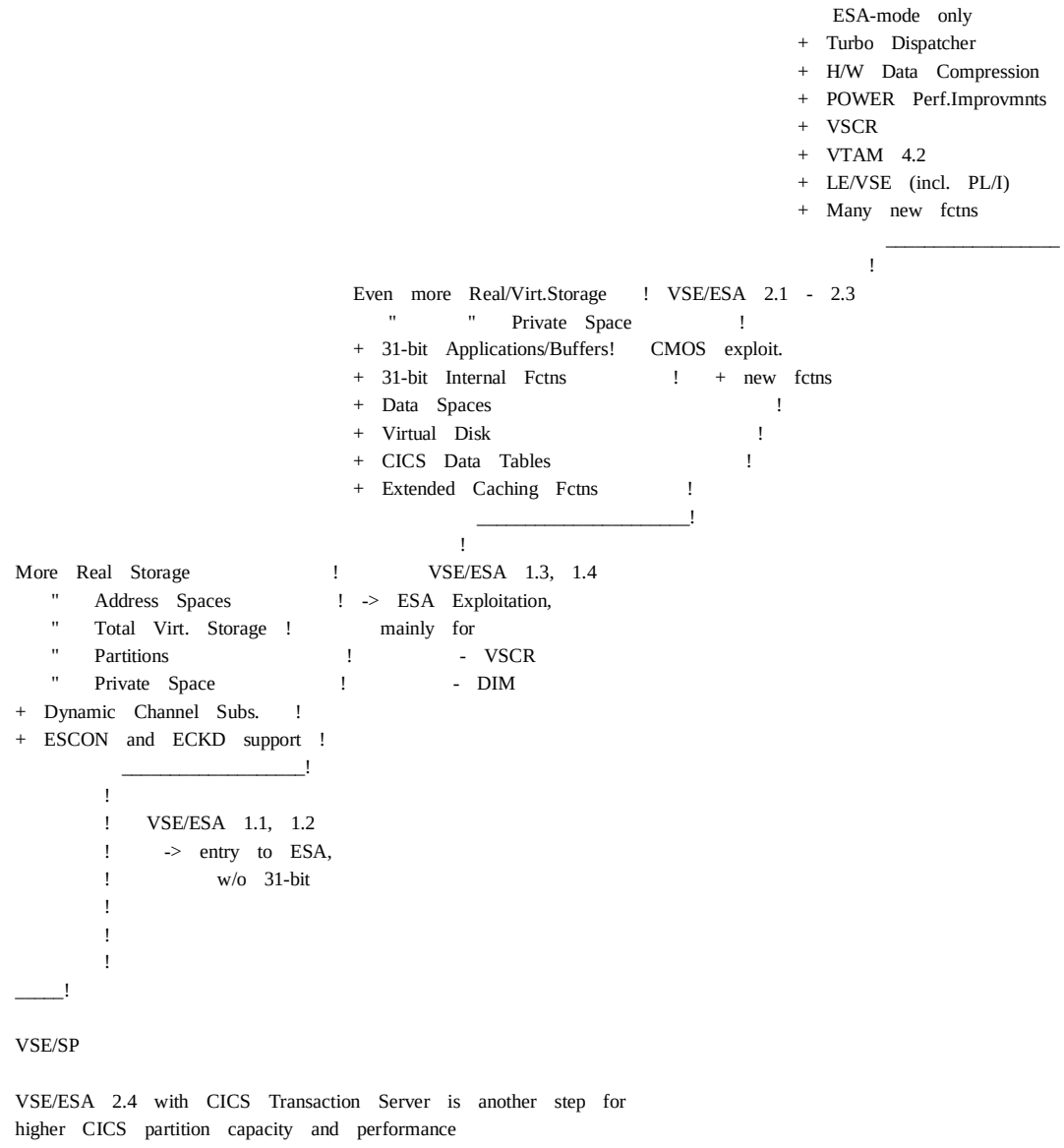
- Hints and Tips for VSE/ESA, by VSE Development,
  2nd edition 01/96, 144 pages
  3rd edition 12/97, 236 pages
  Contained e.g. in the VSE/ESA information on the Internet

- IBM VSE/ESA Hints und Tips, Release 2.2.2 (In German),
  Sept. 97, by Joerg Haertel and Friedrich Hahn, 62 pages
  Available for your IBM representative via
  TOOLS SENDTO STUTVM2 SBZINFO SBZINFO GET ESA22PR LIST3820

Refer also to the VSE/ESA Home page on the Internet:

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

VSE/ESA Performance/Capacity Evolution



VSE/ESA 2.4 with CICS Transaction Server is another step for higher CICS partition capacity and performance

VSE/ESA 2.1 Performance Line Items

This list includes items with performance impact in limited areas

- **Hardware Support**
  - **Turbo Dispatcher and support for the S/390 Parallel Enterprise Servers 9672-Rx1**

    Discussion of n-way processor support is contained in a separate document listed above
  - **ESA Supervisor only**
  - **H/W Data Compression support**
  - **3990-6 Extended Functions support (non-VSAM)**

    (VSAM support already included in VSE/ESA 1.3 by PTF)
  - **Support of RAMAC Array DASD/Subsystem**

    (More exploitation of RAMAC capabilities in VSE/ESA 2.1)
  - **Automatic Cartridge Loader improvements**
  - **Support of Pinned Data for cached 3990s**
- **POWER Performance Improvements**
  - **Maximum DBLK size of 64K**
  - **Increased Output Buffer size**
  - **Increased maximum PNET Buffer size**
  - **Improved Q-entry locking for non-shared POWER**
  - **Enhanced Viewing of Q-entries**
  - **Parallel Browse of Q-entries**
  - **New Output Segmentation macro IPWSEGM**

VSE/ESA 2.1 Performance Line Items ...

- **VSE VSCR for Higher Capacity and Easier Setup**
  - **Major VSAM access control blocks moved above the line**
  - **Move-mode transients in 31-bit SVA**
  - **PL/I applications above the line via LE/VSE**
  - **VSE/OCCF no longer requires a shared partition**
  - **Resulting shared spaces and VSCR**
- **Other Items**
  - **Label Area in native data space**
  - **Improved Console Support**
  - **Page manager enhancements**
  - **NOPDS option**
  - **More ECKD support (FETCH/LOAD, HardCopy file)**
  - **Tracing and Debugging enhancements**
  - **C/S enhancements impact on VSE load**
  - **Miscellaneous items**
  - **LE/VSE performance aspects**
- **CICS/VSE 2.3**
  - **CICS/VSE 2.2 performance benefits preserved**
  - **Support of LE/VSE**

VSE/ESA 2.2 Performance Line Items

- **VTAM 4.2 with 31-bit Support for VSCR**
  - **Private space**
  - **Data spaces**
  - **Total storage requirements**
  - **Data compression**

VSE/ESA 2.2 Performance Items

- **3590 Tape support**
- **REXX CPU Monitor**
- **Turbo Dispatcher**
  - **Relative VSE SHAREs**
  - **Parallel POWER option**
- **VSE/VSAM**
  - **Extended User Buffering**
  - **IKQCPRED tool now standard**
- **New optional products**
  - **SQL/DS 3.5**
  - **LE/VSE 1.4**
- **Performance PTFs**

  POWER, SIR RESET, LE GETVIS, LANRES

VSE/ESA 2.3 Performance Items

Refer to separate part in this document

VSE/ESA 2.4 Performance Items

Refer to separate part, currently in new CICS TS document

VSE/ESA 2.1/2.2 Basic Pathlengths

Ù Possible CPU-time increase (ITR degradation)

With the introduction of many new VSE functions in VSE/ESA 2.1.0 the basic VSE pathlengths have been impacted to some extent as compared to VSE/ESA 1.3.
This may increase the CPU-time of a CICS transaction or a batch program, and may vary depending on a customers workload.

Considering an 'apples to apples' comparison between 2.1 and 1.3 (i.e. w/o exploitation of new functions or parameter values),

VSE/ESA 2.1.x may use overall around 5% more CPU-time

- depending on environment
- without exploiting new features

CPU-time per txn/batch job can be reduced by:

- all those ESA exploitation items documented already for VSE/ESA 1.3/1.4:

  By more VSCR or more 31-bit applications in VSE/ESA 2.1, even more DIM is possible

- VSE/ESA 2.1 specific specific tuning possibilities:

  - POWER spooling/printing/viewing

  - PL/I 31-bit transactions for more VSCR/capacity

  - DL/I CI-size for 'index component'

  - Label Area in native data space

í VSE/ESA 2.1 is a 'CMOS MP exploitation and new function' release

Naturally, it will also run adequately on e.g. bipolar 9121s.

VSE/ESA 2.1/2.2 Basic Pathlengths ...

VSE/ESA 2.2 Pathlengths

Measurements of VSE/ESA 2.2 have shown:

around 1% CPU-time increase vs VSE/ESA 2.1

VSE/ESA V2 on non-optimal ESA processors

- **VSE/ESA V2 is ESA-mode only**
- **VSE/ESA 2.1 exploits ESA architecture even more than VSE/ESA 1.3**

  PC/PR instruction usage

  Program Call/Program Return for console communication, data spaces...

  BAKR/PR instruction usage

  Branch and Stack/Program Return for VTAM session setup and communication

  More Access Register usage

  XPCC now uses access registers, since ESA only

- **ES/9000 processors (incl. IBM CMOS processors) have optimal ESA implementation**
- **Non-optimal ESA/370 or ESA/390 processors**

  (e.g. 4381-9xE)

  need more CPU-time with VSE/ESA V2

  ESA implementation in u-code.

  Starting with VSE/ESA 2.4, ESA/390 is required

PART A.

H/W Support

VSE/ESA V2 n-way support (Turbo Dispatcher)

is discussed in a separate document

ESA Supervisor Only

- **MODE=VMESA throughput is limited by single address space (virtual storage constraints)**

  Single private space as precious as shared space (below the line)

- **VTAM 4.2 requires data spaces, not available else**

  Only small server machines (BTAM only) would have still been possible

- **VSE CCW translation was done anyhow for MODE=VMESA in VSE/ESA 1.3**

- **Deltas between VSE/ESA 1.3 supervisors**

VM/VSE ITR ratios (VSE/ESA 1.3)					
	MODE=VMESA V=V DED/MIN	MODE=ESA V=R/F (Base) DED		MODE=ESA V=R/F MIN	MODE=ESA V=V DED/MIN
	-----*-----> -----**----------->---------->				
9221	0.77 (base)	1.0 (+28%)		0.70 (- 9%)	0.68 (-12%)
9121	0.84 (base)	1.0 (+19%)		0.74 (-12%)	0.72 (-14%)
PACEX I/O intensive workload Deltas for average I/O intensive loads much smaller DED All disk I/O devices DEDICATEd MIN All disk I/O devices MINIDISK (full or partial)					

- **Up to about 19%/28% ITR benefit for DEDicated devices and reserved storage (*)**

- **Up to about 13% ITR degradation for other cases (**) (including full pack minidisks)**

Data Compression

General Alternatives for Implementation

Data Compression could be implemented ...

- **On host processor level**
  - **via compression routine in software**
  - **via compression routine in hardware**

    Called by a single S/390 instruction
- **On controller (subsystem) level**
- **On device level**

- **The place of implementation determines the principal benefits**

Terminology

For Data Compression the following terminology is used:

'Compression' is used both as general term and as specific term for the compression step only

'Expansion' is used to designate the activity of de-compression which is required to read compressed data

Sometimes, compression is mistakenly confused with compaction:

'Compaction' is the process of packaging data more tightly, without trying to use a more efficient coding scheme (just removing interblock gaps or trailing blanks)

H/W Support for Data Compression

H/W Assist

Ù New CSRCMPSC service for compression and expansion

- Invokes the CMPSC hardware instruction (op-code 'B263'), if available (on all newer ES/9000 processors and Enterprise System Server/390s)

- Invokes a fully compatible software compression otherwise

- Displays compression type at IPL time (msg IJ76I)

- Implemented algorithm is based on Lempel-Ziv (LZ), much more powerful than e.g. RLE (Run Length Encoding). (Refer to 'ESA/390 Data Compression', SA22-7208)

- Technical details for IBM personnel is contained in

  'ES/9000 Hardware Assisted Data Compression',
  Technical Reference and Applicability Guide,
  04/93, 21 pages.
  CMTG9000 PACKAGE on MKTTOOLS tools disk
  (IBM Internal Use)

Scope of Applicability

Ù VTAM line transmission

Refer to VTAM 4.2 section

Ù VSAM files

- Refer to the following charts ...

- For even more information refer to

  'VSE/ESA Exploitation of Hardware Data Compression',
  09/94, by Horst Sinram and Dan Janda Boeblingen/Endicott,
  VSE/ESA Technical Conference Philadelphia/Brussels.

  As TC94DCMP Package on IBMVSE TOOLS disk.
  Get a copy via your IBM representative.

  Refer also to an update of this presentation,
  held at the VSE/ESA Technical Conference Atlanta March 95.

VSAM H/W Compression Applicability

„ VSAM KSDS, VRDS, and ESDS datasets,

KSDS with records stored in a CICS Data Table are also eligible to reside on DASD in compressed format. BUT, since CICS Data Tables for KSDSs contain uncompressed data, essentially only DASD space benefits apply.

In case of ESDS, only VSAM record ADDs are possible

No record updates allowed, seen from a VSAM perspective, even if same uncompressed record length is used

„ Not for RRDS (use VRDS instead), not for Alternate Index files

„ Not for VSAM managed SAM

RECORDFORMAT in DTF or DEF CLUSTER

„ Extremely small VSAM files are not suited for compression:

- About the first 8-64 KB of data are non-compressed ('interrogation phase' including 'dictionary mating'), except if data set is REUSEd

- DASD savings would be small, and virtual space overhead as big as with large files

„ MOVE- and LOCATE-mode

- LOCATE mode (not possible for CICS) is also allowed, since extra buffer is used for expanded 'open' records

„ NSR and LSR Buffering

VSAM H/W Compression Applicability ...

VSAM Scope of Applicability (cont'd)

- **Cluster must be defined with the COMPRESSED parameter (no IMBED or REPLICATE allowed)**

  These options are not useful for cached DASDs

- **Specific cases of 'dummy load'**

  Refer to VSAM compression hints if application leaves initial load mode directly after e.g. 1 dummy record

- **Files processed in CNV-mode CANNOT be compressed (by VSAM)**

  CICS TD, TS Aux, DL/I data component, SQL/DS files ...

  'Application' would have to use compression by its own, and specify MACRF=CMP.

  Contact VSE DL/I Development if you have an urgent need to use H/W compression for DL/I ESDSs ('data component')

- **Only data that follows the prime key is compressed (KSDS)**

- **Only data >40 byte are compressed**

- **VSAM RBAs**

  RBA is a VSAM internal item and is not useable/known for compressed files:
  Must be calculated by VSAM, the application CANNOT calculate it

- **No modification required in correct applications**

VSAM H/W Compression Costs

- **CPU-time for compression and expansion alone, i.e. w/o application etc ...**

Relative CPU-time per (uncompressed) byte			
Method/Implementation	Compression	Expansion	
ES/9000 H/W Compression * (LZ algorithm)	11	2.5	
ES/9000 S/W Emulation (LZ algorithm)	50	12	
(Huffman S/W Compression)	(12)	(24)	

- All values are approximate values and would also apply to non-VSAM

LZ Lempel-Ziv compression (compression more expensive than expansion)

* ES/9000 processor models, 9672's similar

- **CPU-time ratio: about 1:3 to 1:5 (11:50 or 2.5:12), much less than for equiv. S/W implementation**

- **Compression cost is higher for initial file loads, since compression dictionary must be built from the first 64K of the file ('initial scan').**

This is not done for REUSEd VSAM files, since compression dictionary is also reused.

VSAM H/W Compression Costs (cont'd)

Ù **Some virtual space for compression code, dictionaries and control blocks**

- Static code and data:

Compression code	IKQVCS	82K	
	IKQVCCS	19K	
	IKQVRCPI	5K	
417 Dictionary Building Blocks (DBBs)	IKQDBB01	192K	

about 300K in SVA-31
\+ about 4-8K in SVA-24

- File space (areas, dictionaries), per compressed file:

	Space in GETVIS-31	
	During OPEN	After OPEN
First file	160K	120K
Others	128K	120K
- Values vary a lot with file data		

VSAM H/W Compression Benefits

VSAM H/W Compression Benefit Overview

- **Performance benefit areas**
  - **'Virtually bigger CIs' have impact on ...**
    1. **Required DASD storage**
    2. **Cache hit ratios**
    3. **Channel traffic**
    4. **Required central (real) and virtual storage**
    5. **Number of DASD-I/Os per tx, Tx response time**
    6. **Maximum file size**
    7. **Elapsed time for file Load, Browse, Backup, Restore ...**

VSAM H/W Compression Benefits

Ù 1. Reduced DASD storage required for compressed files

Depending on compression factor, about half the space required

VSAM Prediction tool IKQCPRED available

Computes compression ratios for VSAM files by an actual compression trial.

Standard in VSE/ESA 2.2, available with PTF UD50096.

Contact your IBM representative and request the program in case you do not want to apply that PTF.
It is
- part of the IKQCPRED PACKAGE on the IBMVSE tools disk, obtainable via the command

  TOOLS SENDTO BOEVM3 VMTOOLS IBMVSE GET IKQCPRED PACKAGE

- available as IKQCPRED.ZIP on the Internet (same URL as this document)

For more info refer also to
VSE/ESA Enhancements, Version 2.2, SC33-6629-00, 12/96, p63.

Ù 2. 'Virtually bigger DASD cache' may increase hit ratios

About twice as much data fits in cache or buffers.

This benefit does not only apply to any DASD cache in cached I/O subsystems, but also to the effectiveness of VM Minidisk Full Track Caching in VM/ESA 1.2.2 and newer releases

Ù 3. Reduced channel traffic for compressed files

Depending on compression factor, fewer MBs have to be transferred.

Biggest reduction is for sequential access, where all transferred bytes are used by the application.

VSAM H/W Compression Benefits (cont'd)

Ù **4. More data records stored per MB/real or virtual (buffer size)**

Note that first the Base KB overhead (virtual) must be compensated, as shown earlier.

í **Less total vitual storage only for large buffer space**

Important for performance behavior:

Unit of compression is the logical VSAM record:

„ **Actual expansion occurs at record GET**

Í Only required records are expanded

„ **Actual compression occurs at record PUT**

Í Updated record may have different physical size, potentially causing CI splits even for fixed record sizes.

Additional freespace may be required for files with frequent update-in-place

VSAM H/W Compression Benefits (cont'd)

Ù **5. Impact on number of DASD-I/Os and response times**

Number of I/Os:

MB virt/real	DIM	->#I/Os	
same		more	-> less
less		same	-> same

Transaction response times:

í **Potentially better**

- vs equivalent S/W compression

via saved CPU-time

- vs uncompressed, if processor power left

via saved I/Os

I/Os are saved when more Data In Memory is used, or when file access is predominantly sequential

Ù **6. Maximum VSAM file size (4 GB) less restrictive**

More records per file

A 4 GB compressed file may now contain much more records, depending on the compression factor
(ESDS and KSDS e.g., RBA used internally by VSAM)

Tuning potential

For files with heavy insert activity...

you may benefit from the ability to specify more CA-Freespace
for such files, in order to avoid expensive CA-splits
(CI splits are much less I/O expensive than CA splits)

VSAM H/W Compression Benefits ...

VSAM H/W Compr. Benefits (cont'd) & Claims

- **7. Sequential file access-times may improve.**

  May give better Elapsed Time on fast processors

  For all 'device rotation bound' activities, the Elapsed Time may improve, since more data records can be transferred per rotation:

  - LOADing
  - BROWSE
  - BACKUP
  - RESTORE ...

IBM Performance Claims

- **DASD Space Savings**

  'often in the range of 40% to 50%'

  for compressed files

  - **Compression factor of 1.66 to 2.0**

- **CPU-time for Compression/Decompression alone**

  'LZ H/W implementation on IBM machines is 3 to 5 times faster than S/W simulation of the same LZ algorithm'

  This claim holds for ALL IBM processors with H/W compression

VSAM H/W Compression Performance

CPU-Time Performance Expectations

- **Dependencies**
  - **Usage share of files defined 'COMPRESSED'**
  - **Actual compression ratios**
  - **R/W ratios**
  - **Intensity of access to compressed files vs other CPU processing**

    This is THE salient parameter for CPU-time overhead

- **CCOST tool for MVS available to IBM personnel**

  Requires essentially as input:

  - Total CPU-time w/o compression
  - Number of GET and PUT operations
  - Avg record length of data
  - Actual/expected compression factor of files

- **Principal classes of compression overheads/workloads**

Type of activity	R/W Ratio	KIcompr (H/W compr.)	Approx. CPU-time increase
No compression involved any	-	0%	
Online transactions (OLTP) 7:1 10	- 50	2-10%	
Batch	3:1 2	- 10	5-25%
Query Read only	.2 - 2	25-200%	
Utility (if compr. involved) Read or .1 - 2 (READ/WRITE, no pgm logic) Write only	50-250% or 1:1		
KIcompr Average # of thousands of (avg) instructions between 2 successive CSRMPSC calls: 'Relative (compressed) record intensiveness'. Refer to discussion below - Only 'compressed' READs or Writes count here - Above ranges similar for MVS and VSE environments			

VSAM H/W Compression Performance ...

H/W Compression with Utilities

Utility / Function	Source file	Compr./Expans.	Target file	
VSAM B/R BACKUP	uncompr.	- / -	uncompr.	
	compr.	- / -	compr.	
VSAM B/R RESTORE	uncompr.	- / -	uncompr.	
	compr.	- / -	compr.	
REPRO & EXPORT/IMPORT	uncompr.	- / -	uncompr.	
	compr.	- / X	uncompr.	
	uncompr.	X / -	compr.	
	compr.	X / X	compr.	
- Source and/or Target on DASD and/or TAPE				

For VSAM B/R BACKUP and RESTORE no compression/expansion is required, so no additional CPU-time is required.

REPRO from compressed source to compressed target includes expansion and compression, since the unit of transfer still is an 'uncompressed logical record' (VSE/VSAM IDCAMS).
Source and target files may even have different compression dictionaries.

EXPORT/IMPORT with compressed data is only possible in RECORD-mode, so similar to REPRO

Refer to the REPRO measurement results

VSAM H/W Compression Results

„ Measurement Environments

9672-R11 processor with cached 9345 DASDs

VSE/ESA 2.1.2 with VSAM patch for DY43952

2 types of workloads: REPRO and Online Transaction mix

Online Transaction Mix Results

1 CICS with 350 active users and 10 sec thinktime

RAMP-C like applications, but different mix and setup:
- heavier mix
- EXPLORE/VSE active

„ Online Mix Results (Summary)

About 26 tx/sec and about 55% to 60% CPU utilization

Case	CPU-time/tx	IO/tx	RT	tx/sec	
Standard Disp.	+5%	-3%	-10%	+-0%	
Turbo Disp.	+4%	-5%	-13%	+-0%	
- RT benefits may not show up at very high utilizations					

í CPU-time increase as expected

í Small I/O benefits here

VSAM H/W Compression Results ...

REPRO Results

Very compression/expansion intensive load

KSDS file with 166667 250-byte records (incl. 10 byte key)

Data-CISIZE=8K, compression ratio = 1.9 (about 'average')

Turbo Dispatcher was used

„ REPRO Measurement Results (Summary)

REPRO with H/W Compression				
	Uncompr. ->Uncomp.	Uncompr. ->compr.	Compr. ->Uncom.	Compr. ->Compr.	
BUFND=40 (optimal I/O setup)					
Elapsed time 1.00	1.41 (Base)	1.14	1.52	
CPU time 1.00	2.69 (Base)	1.63	3.29	
CPU-utilization 33.5%	63.9%	48.1%	72.7%	
BUFND=4 ('normal' I/O setup)					
Elapsed time 1.00	1.22 (Base)	1.03	1.25	
CPU time 1.00	2.74 (Base)	1.68	3.33	
CPU-utilization 24.5%	54.7%	39.8%	64.9%	
- Run parameters as described above				

í CPU-time increases as expected

í No elapsed time benefit here

Less total I/O time saved here than increase in CPU-time was on this processor

VSAM H/W Compression Prediction

CPU-Time Performance

„ 'Relative (compressed) record intensiveness'

KIcompr = avg. # of thousands of (avg) instructions between 2 successive CSRCMPSC calls

= total number of K instructions w/o compression / # compressed or expanded records

- KIcompr is the higher the more application logic is done

- Exploitation of DIM improves run-times and CPU-times
  by reducing physical I/Os,
  but does NOT reduce the number of reads of compressed records

- VSE/VSAM does NOT call CSRCMPSC if a record is NOT compressed
  or NOT eligible for compression.
  Situations that compression is called but later on record
  is stored uncompressed can be neglected for estimates,
  also since R/W ratio normally is about 4:1 up to 9:1
  (%R=.8 to .9)

„ CPU-time overhead for H/W compression

Relative CPU-time Increase (incl. S/W setup)

CPUTdelta = (Expansion + Compression cost) / Base cost w/o compression

= ((%R x KIexpan) + ((1-%R) x KIcompr)) / KIcompr

%R = share of READs = (R/W) / (R/W +1)
1-%R = share of WRITEs of compressed records

KIcompr = Base compression + #bytes/rec x xxxKI
(about 0.2..0.4) (about 0.011)

KIexpan = Base expansion + #bytes/rec x yyyKI
(about 0.2..0.4) (about 0.003)

VSAM H/W Compression Prediction ...

CPU-Time Performance (cont'd)

„ **Sample tx-workload caclulations**

Worst case assumption: All files compressed

Workload	KI/tx	Total IO/tx	READ/tx WRITE/tx	Log. R/W rat.	Log. RECL %R	Avg KIcompr (byte)	CPU-time w/okey	Approx. increase: CPUTdelta	
RAMP-C	225KI	9.4	16.2	2.35	4.0	200	16.8KI	8.2%	
RAMP-C DIM	190KI	"	7.2	"	"	"	14.2KI	9.7%	
DSW	220KI	6.3	4.7	3.31	1.6	200e	31.9KI	4.0%	
YOUR W/L	___KI	_._	___._	_.__	_._	___	__._KI	___%	

CPUTdelta for H/W compression =

$$\frac{\%R \times (0.3 + RECL \times 0.003) + (1-\%R) \times (0.3 + RECL \times 0.011)}{KIcompr}$$

- Figures are first estimates only, may be refined
- KI = K instructions
- Values hold for ES/9000 processors
- Note that RAMP-C is very file intensive

Note:

To obtain approximate effective pathlength/avg tx,
take total CPU-time per avg tx and multiply by an approximate
S/390 MIPS value, which is not so critical for these estimates.

Your IBM representative may assist to do such calculations

3990-6 Control Unit Support

3990-6 Exploitation for VSE/ESA 2.1

For a general description of 3990-6 performance refer to
'VSE/ESA I/O Subsystem Performance Considerations'

- **Record Cache Mode support**

  VSE/VSAM

  SQL/DS and DL/I data (includes VSE/ESA 1.3.x)

- **Adaptive Caching support**

  3990-3 support mostly sufficient (but 3990-6 support 'faster')
  (includes VSE/ESA 1.3)

- **RDF set in ECKD channel programs**

  (Regular Data Format)

  Required for Record Cache Mode, beneficial for Adaptive Caching

  VSAM, represents 70 to 80% of all I/Os

  Medium potential, already in VSE/ESA 1.3

  LIBRarian and FETCH/LOAD

  Some potential, overall

  Page Manager

  Small potential, if paging

  Lock Manager

  Small potential, since update occurs shortly after read

  HardCopy support

  Was CKD in VSE/ESA 1.3 with 2K blocks, is 4K in 2.1.
  Small overall potential by RDF

- **Support of 3990-6 'Enhanced Mode'**

VSAM APAR/PTF for VSE/ESA 1.3 is DY43072/UD90363

If under VM/ESA 1.2.2, make sure VM APAR VM59317 (PTF UM27166) is applied.

PART B.

VSCR for Increased Capacity (incl. DL/I)

VSE VSCR for Capacity

- **Private space below 16M still a precious resource**

- **Targets:**

  1. **Provide better exploitation of a given private space**

  2. **Reduce shared space requirements below 16M**

These are 2 cumulative aspects which apply to VSCR.

- **Higher CICS capacity by VSCR, both for 24-bit and 31-bit transaction environments**

- **As far as possible, new function code resides in SVA-31:**

- **SVA-31 usage increased from 40K to several MB**

VSE/ESA 2.1	VSE/ESA 2.2	VSE/ESA 2.3	VSE/ESA 2.4	
2200K			2905K	2919K

The following VSE/ESA V2 foils show our efforts

- to compensate for increasing shared space below the line (caused by the implementation of new functions)

- to allow for further growth

VSAM and PL/I Private Space Relief

Better exploitation of a given private space

Ù Major VSE/VSAM access control blocks above the line

The following VSAM control blocks were moved above the line

Control Block	For	Number	Size (byte)
PLH	Placeholder	NSR 1 per file string LSR 1 per pool string	per NSR 656 * +maxkeyl. per LSR
BCB	Buffer control block NSR,LSR 1	per buffer	104
BHD	Buffer header NSR,LSR	2 per PLH	48
BSPH	Buffer subpool header LSR 1	per subpool	74
RSCB	Resou.sharing c.block LSR 2	per PLH	16

- Sizes shown hold for VSE/ESA 2.1
* PLH slightly increased vs 1.3/1.4

í Savings vs VSE/ESA V1.3/1.4

about 1K x 'string_number' + 0.1K x 'buffers'

Ù 31-bit PL/I applications via LE/VSE 1.1

Batch programs and CICS transactions

Also DL/I applications

Refer to these subjects

DL/I 1.10 Performance Enhancements

Ù DL/I parameter lists may reside above the line

APAR PN67649 (PTF UN73450) for VSE/ESA 1.3 and up

„ Parameter list resides where application is

- I/O area for data (about 200 byte)
- SSAs, 1 to 15 per DL/I call, 200 byte each

-> About 1K per task (DL/1 transaction)

- is copied from above to below the 16M line once per DL/I call ('MVCL')

„ Applies to 31-bit languages

- COBOL (II and 'LE')
- PL/1 for VSE/ESA
- High Level Assembler

„ Performance Results

No measurable impact on CPU-times by additional MVCL instructions

Holds both for CALL and for HLPI interface.

Performance runs were done with PL/I and LE/VSE

Ù Program isolation control blocks above the line

APAR PN88972 (PTF UN96346,UN99402) for VSE/ESA 1.3/1.4 and up

Control blocks for record locking (PI-enqueuing elements)

DL/I 1.10 Performance Enhancements (cont'd)

Ù Improved DL/I Statistics

This update avoids misinterpretation of statistics for performance tuning reasons, refer to APAR PN83042 (PTF UN93565)

Ù DL/I Index KSDS CI-size up to 30K

APAR PN68583 (PTF UN97235/UN97236) for VSE/ESA 1.3/1.4 and up.
Information APAR II08400

Former size was 4K only

Ù More buffers per DL/I subpool

APAR PN89468 (PTF UN97235,UN99402) for VSE/ESA 1.3/1.4 and up

„ Up to 255 buffers per DL/I subpool

Still all buffers (size = ESDS CI-size) reside below 16M.

Default value is the previous maximum value of 32 buffers.

The number of buffers per DL/I subpool should not be too extreme, due to buffer search overhead

„ Exploitation requires changed JCL or ACT

Via HDBFR=(nn, ...) in DL/I parameter card or
'DLZACT TYPE=BUFFER...' statement for ACT generation

For reasonable exploitation of DIM, refer to the VSE/ESA V1.3/1.4 performance document.

DL/I 1.10 Performance Enhancements (cont'd)

Ù DL/I PSBs above the line (PSBLOC=ANY or GV24)

APAR PQ09904 (PTF UQ10966) as of 11/97,
APAR PQ29613 (PTF UQ33505) as of 08/99

Program Specification Blocks may reside above the 16M line in VSE GETVIS-31:

Specify PSBLOC=ANY or =GV24 in DLZACT TYPE=CONFIG

When an online request with DL/I is scheduled by CICS, the PSB is temporarily copied from GETVIS-31

- into the CICS/VSE DSA (PSBLOC=ANY)
- into VSE GETVIS-24 (PSBLOC=GV24)

The new PTF with GV24 was required since for frequent syncpoints, CICS/VSE DSA fragmentation could occur, potentially resulting to SOS conditions.

In such cases, specify PSBLOC=GV24.

The VSCR for both cases is identical and may be calculated as
#PSBs x avg size per PSB
e.g. 100 x 2..4K = about 300K

(PSB size varies with parameters, mostly with #PCBs)

DL/I 31-bit Exploitation (VSCR and DIM)

	VS/COBOL PL/1-24	Assembler	VSE COBOL a)	II Assembl.	HL- b)	PL/I	
DL/I code	-		-	-		-	
DL/I application	-	-	X	X		'LE'	
DL/I index comp. + HISAM data (CI-size =4..30K) APAR PN68583	X		X	X		X	X
DL/I data comp. (ESDSs) (CI-size <=30K) (>32 buffers) APAR PN89468	-	-	-	-		-	
Parameter lists APAR PN67649	-		-	X		X	'LE'
DL/I PI control blocks (pgm isol) APAR PN88972	X	X	X	X		X	
DL/I PSBs APAR PQ09904 and PQ29613	X	X	X	X		X	X

X means 'above-16-MB-eligible' (VSE/ESA 1.3 and up)

a) COBOL for VSE (LE/VSE, only VSE/ESA 1.4 and up)
b) PL/1 31-bit (LE/VSE, " " ")

- Table holds for DL/I Batch and CICS

- Table holds for CALL and HLPI interface
(HLPI not available for assembler languages)

- Pre-req to have DL/I VSAM buffers above 16MB:

HSMODE=ANY in Batch DL/I control statements
in Online DL/I DLZACT subparameter TYPE=CONFIG

- DL/I data component files are VSAM ESDSs with UBF and CNV, thus not directly suited to reside above

- Refer also to DL/I information APAR II07101 which contains general information about DL/I 1.10 PTFs

- DL/I 1.11 allows up to 20G per DL/I data base (more details in CICS TS performance document)

General DL/I Performance Aspects

General DL/I Data Base Performance Aspects

„ VSAM User Buffering and CI-processing is used (UBF,CNV)

Í DL/I uses logical GETs and PUTs for any processing, resulting in 1 CI (and 1 I/O) per VSAM request

„ ESDS CI-size can be up to 30K

„ DL/I segment updates (INSERT, REPLACE, DELETE)

May cause that all segments of a single data base record (represented by its root segment) require additional space in a new CI, not physically adjacent to the original one(s):

Í increased scattering of CIs

Logical sequential processing (incl. UNLOAD) needs more and more random physical accesses

„ Avoid blocking of higher priority tasks

When using fast caches (OPTI CACHE or VM MDC), DL/I requests could monopolize a CPU. To avoid, make sure the PTFs for APARs DY44486/44626 were applied which uses VSAM EXCPAD in all cases.

„ Reorganization via UNLOAD/RELOAD

Logical backup/restore of DL/I records (via GET/PUT CI)
(1 CI per VSAM I/O)

„ No reorganization via IMAGECOPY/RECOVERY

Backup/restore of CIs in physical sequence
(BUFND CIs per VSAM I/O, BUFND specified e.g. in user DLBL)

With respect to Cache bit settings in the ECKD channel programs, refer to 'VSE/ESA I/O Subsystem Performance Considerations'

Refer also to the CICS DL/I Statistics Checklist in
' IBM VSE/ESA Hints for Performance Activities'

Shared Space Impact Factors

The following VSCR items have impact on shared space requirements below the line:

- **All Move-mode transients in 31-bit SVA**

  About 26 KB moved above the line by this item

  (MOVE-mode transients $$BA, $$BC, $$BO, including 5K VSAM.
  The $$BOCRTx transients (14K) vanished)

- **VSE Dispatcher loaded directly above the supervisor**

  Turbo Dispatcher of about 24K

  'Standard' dispatcher of about 3.5K

  Í Turbo Dispatcher costs about 20K shared space 24-bit

- **Resident 3480/3490 ERP in supervisor (2.1) or SVA-31 (2.2, IJBTAP phase, includes 3590)**

  No CPU-time impact, just shared space

  New SGTAP macro for ERP

  Only 1 out of 9 tape ERP transients still required:
  $$ABERAY out of $$ABERAU-Z, $$ABERTX-Z

  Í Reduces usage of PTA

Space survey	ESA 2.1	ESA 2.2	
SGTAP code		+20K	SVA-31
ERP data areas	+18K supvr	+2K supvr	
All deltas vs VSE/ESA 1.3/1.4			

Shared Space Requirements ...

Ù **VSE/OCCF no more in shared partition, Message handling code in SVA-31**

Private partition

Required for unattended node support and message routing to NetView

Supervisor appendage

Sufficient for message translation, automation and command and reply translation
(started and stopped by AR commands)

	OCCF 1.4.0	OCCF 1.5.0	
		Private Partition	Supervisor Appendage
SVA-31	- -	$IJBOCFS 13.5K - + Tables	
SVA-24	-	$IJBOCCF 8.5K	
Private partition - -24	$IJBOCCF - +NetView appe. - - +Tables - - in GETVIS -		
Shared partition OCCF code -24 +	- NetView appendage - + Message buffers - + Tables in GETVIS	- - - - -	- -
Total shared-24	80K 8.5K	8.5K	

- No code/areas had to be moved into the 2.1 supervisor
- Messages now buffered by Console Router

Shared Space Requirements ...

Supervisor Comparison

Supervisor sizes	1.3.6	VSE/ESA 2.1.0	VSE/ESA	Delta	
Total V-size (MAP) (DEBUG=NO, incl. std disp.)		532K	600K	+68K	
Base Supervisor (DEBUG=NO)		318K	379K	+61K	
Some individual deltas					
Tape ERP		-	-	+18K	
DEBUG for IPL		-	-	+4K	
...		-	-		
Adjacent to supervisor:					
Std dispatcher	-	3.5K	-		
or Turbo dispatcher	-	24K	+20.5K		
- $$A$SUPX supervisors - Parameters as listed in table below - New dispatcher 'area' is adjacent to supervisor					

VSE/ESA 1.x/2.x total supervisor sizes still depend on:

- generation options (incl. IODEV)
- SYS parameters

SYS parameter	1.3	2.1	Shared space	cost
BUFSIZE (spec'd)	1500	1500	72 byte each	
(actual)	1533	1550	-	
(max used)	326	357	-	
CHANQ (spec'd)	deflt	deflt	32 byte each	
(actual)	168	163	-	
(max used)	-	68	-	
NPARTS (spec'd)	44	44	<1K each	
SDSIZE	deflt=64K	deflt=64K	-	

- **Net 2.1 supervisor increased by about 60 KB**

Shared Space Requirements ...

SDL and VLA Comparison

(VSE standard performance measurement system in Boeblingen,
after DL/I and COBOL/VS phases were loaded via SET SDL)

	ESA 1.3.6	ESA 2.1.0	Delta	
#SDL-entries (24+31) 326	453	-		
VLA-24				
#entries	317	246	-	
Total space	1718K	1631K	-87K	
incl.				
VTAM part	(41)	(8)	-	
	(170K)	(32K)	-	
VLA-31				
#entries	9	207	-	
Total space	40K	2740K	-	
incl.				
VTAM part	-	128)	new	
	-	(816K)		
HLASM part	-	(10)	new	
	-	(430K)		
- All entries in this table are 'used' values				

More specifics for SVA-24 phases:

Some new phases $IJBSLIB 11.8K (3494-support)
ARXIOLAR 7.2K (REXX)
ARXRX24 5.1K (REXX)
ASMA90 9.7K (HLASM)

Some increased phases $IJBAR + 19K (to 65.5K)
$IJBDCMD + 2K (to 24.5K)
$IJBLBR +22.5K (to 253.8K)
IKQVCLC + 1.0K (to 4.0K)
IKQVMSG + 2.9K (to 7.1K)
IKQVOPEN + 11K (to 60.5K)
IKQVRM + 2K (to 85.3K)

SVA-24 phases wich reduced in size, or which moved above: NOT shown

VSE/ESA 2.1 Space Summary

Available partition space below 16MB

Observations from sample measurements
(Figures will vary with workload and product configuration/setup)
RAMP-C DIM (VSAM LSR), no products with additional shared storage needs

„ Total size of required shared space below the line

		ESA 1.3.6	ESA 2.1.0	Delta	
Supervisor (DEBUG=NO, incl. disp. +SDSIZE)	532K	600K		+68K	
SDL list		24K		28K	+4K
Virtual Library Area-24	1718K	1631K		-87K	
System GETVIS-24 (used)	K	K		K	
Label Work Area	108K	108K		0K	
VPOOL		64K		64K	0K
Shared partitions (SPSIZE)	0K	0K		0K	
Total Shared Used		K		K	K
Reserve **		K		K	-
Total Shared-24		M		M	M
- SDL, VLA and Label Work Area from IUI panel 363					

- Refer to charts for Supervisor and VLA details

** 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:

___ MB (this example)

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

PART C.

Enhanced Label Area and LTA

VSE Label Area in Native Data Space

Label Area on Virtual Disk (VSE/ESA 1.3/1.4)

Savings of physical I/Os to Label Area on DASD

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

were a major performance improvement, but ...

Still had included:

1. **CPU-time overhead (CCW x-lation,'I/O' handling)**

Came along only for the following costs:

2. **CPU-time overhead (VSE Virtual Disk)**
3. **Small CPU-time increase by the non-indexed label search in the FBA design (vs CKD)**
4. **Reduced total number of labels (vs CKD)**

Enhanced Label Area solves items 1, 2 and 4

- **Maps standard FBA label area concept to native use of a data space**
- **Gets rid of I/O (CCW) handling, but keeps major existing code**
- **Extends the Label Area capacity compared to FBA and/or VD**
- **Must be activated by customer (VSE/ESA 2.1 to 2.3)**

  After the switch to the Enhanced Label Area is done, the Label Area on real disk is no more needed until the next IPL

VSE Label Area in Native Data Space ...

Performance Advantages

„ Label Area capacity increase

	VSE/ESA 1.3 CKD/ECKD	VSE/ESA 1.3 FBA/VD	VSE/ESA V2 Enh.Lab.Area
Total #labels (#'FBA'-blocks)	about 9000 -	about 3000 (992)	about 9000 (2880)
# label sub-areas: total (perm/temp)	239 (39/200)		712 (common)
- The Enhanced Label Area pretends to be on VD, BUT uses ESA data space natively			

Up to about 9000 labels

Up to 2880 (512 byte or 'FBA') blocks can be specified to hold the label area. This allows to store as many labels into this area as into the largest label area on a CKD device.

Up to 712 label sub-areas

This is enough to create the theoretical maximum of label sub-areas possible in the current VSE/ESA system: System label, class label for all classes, and up to 3 label types for each possible partition.

Before the VSE/ESA 2.1 Label Area enhancement, the label sub-area situation was as follows:

Permanent 1 std label
26 class labels
12 parstd labels (static part.)

Temporary 200 'free usage labels' (ICCF relict)
or user labels
or dynamic partition-std labels

„ CPU-time reduction

As label data is moved directly from and to the data space, all CPU-time needed to do CCW processing and to simulate virtual disk (VD) is saved

Compatibility with Former Releases

„ Combination of new and old code

The old phase IJBSLA is used only during IPL, except if the user should stay on real disk.
The new phase IJBSLAD becomes and stays active when the label area is switched to a data space.

„ VDISK command still to be used

The VDISK command is still used to create the data space for the label area. This was done to stay compatible with the former release. Still a virtual disk is created with 'cuu-address' but, under the cover, access to the data space is done without using CCWs and without using Virtual Disk code.

Since no SVC0s are used anymore for label area processing, requests to the Label Area are no more seen e.g. by VSE Job Accounting.

„ Space used for labels

As the space for a virtual disk is allocated in units of 960 FBA blocks, it is possible to allocate 960, 1920, or 2880 units of 512 byte areas in this data space. This is enough to handle about 3000, 6000, or 9000 labels.

„ Still full support of labels on real disk

The code that handles labels on real disk is still contained. Therefore a DLA command is still needed during IPL and labels used during IPL-time (e.g. for IJSYSRS.SYSLIB) can be written into the label area on real disk before the VDISK command is given.

It is also possible to stay completely on real disk even though this is not advisable any more.

It is no more possible (and required) to use the 'old' method of having the Label Area on a 'real VSE Virtual Disk'.

VSE Label Area in Native Data Space ...

Virtual Storage Requirements

„ SVA-24/-31

- 12K in SVA-24 for static partitions

- 96 byte SVA-31 per active dynamic partition and label type

  (such label group entries for dynamic partitions are only allocated when required)

  Example:

  20 active dynamic partitions, ech using temporary and permanent partition labels

  20 x 2 x 96 = 3840 byte in SVA-31

Enhanced Label Area Measurements

„ Some Performance Observations

Label Area comparison:

Native data space vs real (CKD) disk					
Label Area Intensiveness	Activity	Elapsed Time	CPU-time	#I/Os	
Very low	few	Total batch jobs, files in system & JCL	-0.5%	-0.5%	-1%
Average	some	CICS startup, files as job labels	-20%	-3%	-9%
High	some	CICS startup, files, but many std labels	-50%	-5%	-21%

9221-170 with cached 9345 DASDs

Standard VSE/ESA 2.1 system setup

Refer also to Label Area on Virtual Disk (VSE/ESA 1.3)

VSE/ESA Enh. Label Area Capacity

- **Background Info**
  - **Label Area space is managed in 2K blocks ('allocation unit')**

    Each allocation unit (4 512-byte 'FBA' blocks) can host e.g. up to 19 file labels (each 104 byte)
  - **Maximum total size of the Label Area is BLKS=2880 'FBA' blocks**

    This is 2880/4 = 720 2K allocation units, corresponding e.g. to 720x19=13680 labels of 104 byte (ideal case).
  - **Each label sub-area (up to 712 allowed)**
    - **has its own 2K allocation units**

      not shared with others
    - **can use as many 2K blocks as available**
  - **Allocation units are freed as soon as no labels are contained**

    The list of labels in a label sub-area are 'compressed' by moving the space of a deleted label to the end
- **Conclusion**
  - **Individual label sub-areas are NOT limited in size, only via total space**
  - **Label Area space is well used.**

    When free allocation units are exhausted,
    msg 1L10D LABEL AREA EXHAUSTED is presented

Let us know, if your requirements should exceed the current limit

Logical Transient Area (LTA)

LTA Function

- **LTA is a single threaded VSE supervisor area.**

  It is/was used for serializing VSE-wide services via $$B.. phases, instead of (recommended) LOCK/UNLOCK

  - **Area is PFIXed, a page fault can only occur for related data or services**

LTA Usage Examples

- **OPEN/CLOSE entry and exit**

  for BAM and VSAM files

- **VTAM OPEN ACBs**

- **Vendor products (Disk and tape mgmnt, etc)**

- **EOV processing (especially tapes)**

- **Some AR commands**

  AR commands using LTA via $$BATTNA, calling $$BATTNx:

ALLOC	MPXGNT	PAUSE	START
ALTER	MSG	RESERVE	UNLOCK
DISPLAY	LOCK	SETMOD	VOLUME
MODE	UNLOCK	SIZE	LFCB

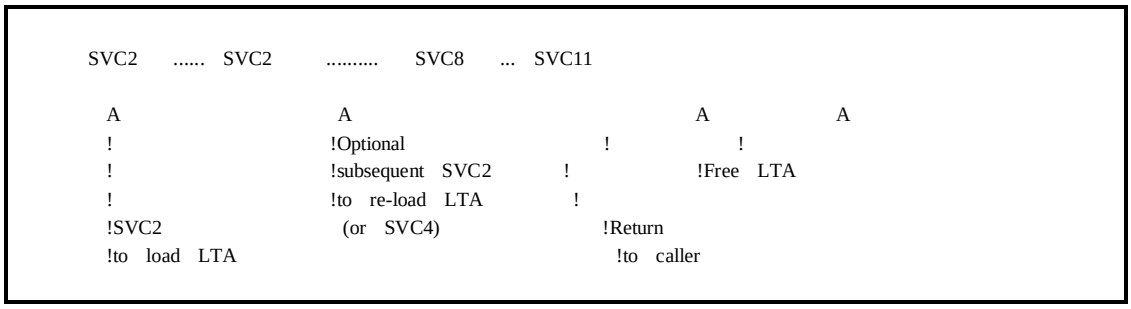
  For AR commands which not or no more require the LTA, refer to a later chart.

Logical Transient Area (LTA) ...

Other (less important) Transient Areas

- Physical Transient Area (PTA) for $$A phases
- Recovery Transient Area (RTA) for $$R phases
- CRT Transient Area (CRTTRNS) for $$BOCRT

LTA Usage Scheme



Not all SVC2s lead to LTA usage. Exceptions are SVC2s for the following B-transients NOT occupying the LTA:

$$BACLOS	$$BEOJ3A	$$BJDUMP
$$BDUMP	$$BEOJ4	$$BPDUMP

More Info

- LY33-9165-00 (Diagnosis Reference Logical Transients V6.1)
- SC33-6336-00 (Diagnosis Reference LIOCS V6.4)

OPEN/CLOSE Processing

Major OPEN/CLOSE processing does NOT occupy LTA

Schematic examples shown here

Ù BAM OPEN/CLOSE

o Load $$BOPEN/$$BCLOSE into LTA (SVC2)
  - Determine type of file (here: BAM, disk or tape)
  - Locate OPEN/CLOSE module in SVA
  - Do a Part. GETVIS for save area
o Transfer control to OPEN/CLOSE
  - Free LTA (SVC11)

  * Do most of BAM OPEN/CLOSE processing,
  * including I/Os to Label Area and VTOC
  * FREEVIS save area

o Exit to exit transient $$BOSVLT or $$BOTLTA
o Return to user pgm (SVC8)
  - Free LTA (SVC11)

Ù VSAM OPEN/CLOSE

o Load $$BOPEN/$$BCLOSE into LTA (SVC2)
  - Determine type of file (here: VSAM)
o Load $$BOVSAM/$$BCVSAM into LTA (SVC2)
  - Locate IKQVOPEN/IKQVCLOS module in SVA
  - Do a Part. GETVIS for save area
o Transfer control to IKQVOPEN/IKQVCLOS in SVA
  - Free LTA (SVC11)

  * Do most of VSAM OPEN/CLOSE processing,
  * including I/Os to VSAM catalogs
  * FREEVIS save area

o Exit to exit transient ($$BCVS02) to terminate
o Return to user pgm (SVC8)
  - Free LTA (SVC11)

LTA Performance Background

ù History

From time to time customers told us that they have a 'problem' with the single LTA (Logical Transient Area):

Due to a higher LTA utilization, LTA BOUND conditions occur (BOUND=81) for other partitions/services also requiring the LTA.

We think that this problem does no more exist in todays VSE releases (VSE/ESA), ...

except the VSE LTA is MISUSED by badly written/designed programs.

It is no question that with the increasing load per VSE system (100 MIPS and more, usually also with increasing number of concurrently active VSE partitions), the LTA usage must be carefully observed, to avoid that it may become a bottleneck again.

In VSE/ESA V2 no more the MLTA feature from FAQS (CA) is available, and, we think, required, except in 'misuse' cases.

ù Performance Observations

Very ancient VSE measurement results (VSE/SP 3.1]]]) done in the lab showed for 4 concurrently active batch partitions (PACEX4):

> only 0.3% of time a partition was waiting for LTA, since LTA already occupied.

We currently have no indications that the LTA is still a bottleneck or is again to become a bottleneck.

If LTA BOUND conditions occur(red) frequently:

> Specific customer applications or vendor products 'misuse(d)' the LTA, especially by issuing I/Os while occupying the LTA, instead of using LOCK/UNLOCK for serialization.

LTA Offload by IBM

Ù Actions done long ago to offload LTA

Most of them already included in VSE/ESA 1.1.

„ Introduced MOVE-mode for many B-transients

Load a transient w/o FETCH I/O to library on disk.
Just move it from SVA to LTA within virtual storage

„ Removed transient services from LTA

E.g. do most of OPEN/CLOSE processing in SVA only

„ Restructured transients such that only in very rare cases LTA is occupied during I/O

„ Implemented all newer AR commands in SVA

New SVA-24 phase $IJBATTN in VSE/ESA 2.2.0 moved more AR commands out of the LTA: (LTA transients $$BATTND and $$BATTN2 no more used by IBM)

LIBSERV	PRTY	SYSDEF
MAP	QUERY	TPBAL

„ Temporarily gave the LTA user highest dispatch priority of all user tasks

Ù Some AR commands not requiring LTA

CANCEL	MSECS	PRTYIO
CACHE	MTC	STATUS
DUMP	OFFLINE	VOLUME
GETVIS	ONLINE	

Measuring/Tracing LTA

Activities that may help to investigate a perceived LTA problem:

„ Observe LTA BOUND conditions

- Do a STATUS command on the console when such situations occur
- Use TD with SIR MON=ON (up to 10% CPU-time overhead) and see the number of LTA BOUND (BND-81) conditions
- Do a (very short) SDAID trace of SVC2 and/or LTA phase fetches

„ Use a VSE System Performance Monitor

Use e.g. EXPLORE/VSE and look at

- Status Event Detail display
- System Summary Report (# LTA loads)

Also, e.g. FAQS DEBUG info may show which task currently owns the LTA and e.g. waits for an I/O completion.

In case you know which task causes trouble, you may CANCEL that task to escape temporarily.

í Find the 'black sheep'

í Make sure you have latest vendor level code

LTA Coding Recommendations

Coding Hints for Programmers on System Level

- **Do not misuse single LTA**

- **Remove/Redesign your $$B transients**

  - frequently used
  - using the LTA very long by issuing I/Os

  - **Don't issue I/Os out of or during LTA occupancy**
  - **Never issue a message out of LTA, waiting for reply (Decision, Action)**

- **Move services from LTA into the SVA.**

  Use LOCK/UNLOCK for serialization of a 'smaller' resource than the global LTA

  Achieve gating by less unsocial means

- **Don't issue SEIZE/RELEASE out of LTA**

  Deadlock potential

More LTA Performance Background

- **LTA no more used by**
  - **System DUMP (PDUMP, IDUMP, ABEND DUMP)**

    Note that in case a task which used the LTA caused a dump, the LTA is blocked during that dump.
  - **End of task (incl. jobstep)**
  - **VSE/POWER avoids/reduces LTA usage**

    If new output segmentation macro is used

    (since VSE/ESA 2.1.0, was $$BSGMNT).
    Refer to separate chart in VSE/POWER part.

    If printer not ready, when FCB is to be loaded

    (since VSE/ESA 2.2 time, was $$BATTF2).
    Does, unfortunately, not remove the problem for all printers.
    Refer to separate chart in VSE/POWER part.
- **Tell us, which services/functions/modules still misuse the LTA and behave unsocial**

  In case of vendor programs, talk to vendor first and then let us know

PART D.

Misc. Items for VSE Base

Improved Console Support

„ Faster display of console messages

Applies to System Console (which is a 'Master' console)

Improved IPL times

No console transients required anymore
(MOVE mode is/was not available during IPL,
thus savings of I/Os to DOSRES)

Faster elapsed times when jobs are console intensive

'Spooling' of console output by Console Router and
Clustering of console I/Os by CRT system task

„ Some Performance Observations

VSE/ESA 2.1 vs VSE/ESA 1.3.6					
Console Intensiveness	Activity	Elapsed Time	CPU-time	#I/Os	
Average	Batch jobs with bigger JCL	-1%e	+1%e	-2%e	
Maximum	Displaying console messages or Redisplay	7x faster	-25% * +20% **	-80%	

* PAUSE=1 sec (default)
** PAUSE=0
- Values above hold for usage of the 'SVC0' I/F
  VSE/ESA 2.1 with 'WTO' requires slightly less CPU-time
- 9221-170 with cached 9345 DASDs
- Standard system setups for VSE/ESA 1.3/2.1

„ Use of PF keys including RETRIEVE function

„ Use of the ES/9000 Integrated Console enabled for safety reasons, though display slow

Improved Console Support (cont'd)

„ Improved performance for the IUI ='CICS' console

Applies to IUI consoles (be it a 'Master' or a 'User' console)

Now as native CICS transaction (IECN),
no more in ICCF pseudo-partition

New long-running CICS transactions IECM and IECA
to start IECN, if one of them is posted

Í You may exclude these 2 transactions from any CICS monitor summary statistics

no more screen refreshes required by the operator

„ EXPLAIN function standard on any full screen VSE console

No overhead,
only a VSAM OPEN of 'VSE.Message.Online' file at VSE startup

„ Improved REDISPLAY performance

- More selective and effective scroll back
  (positioning within Hard Copy file)

„ Performance impact of REDISPLAY and HELP/EXPLAIN

- At REDISPLAY and HELP/EXPLAIN, console messages are kept in internal buffers. If all buffer space for a partition is full, the partition cannot proceed anymore (at about 13 messages per partition).

  This state is called 'console buffer bound', indicated as 'Waiting for router buffer space' in the STATUS command display

- A performance enhancement (APAR PN77905, PTFUN84650) allows that messages for a console in such a status are only recorded in the Hard Copy file. Thus no partition can be delayed by REDISPLAY or HELP/EXPLAIN usage.

Improved Console Support (cont'd)

„ Small VSCR in spite of new console functions

- removed C-transients from SVA-24 ($$BOCRTx, 14K total)

- removed console buffers from dynamic part of SUPVR
  (24x176 byte = 4K)

- space used in SVA-31:
  (all new code is 31-bit)

$IJBCRT	121K	(CRT support)
$IJBCSIO	159K	(incl. router, console I/F, WTO, WTOR..)
$IJBEDEF	5K	(console definitions)
$IJBLSTK	1K	
$IJBPHCF	61K	(Hardcopy support)
$IJBSINA	16K	
$IJBSPDT	49K	(service processor support, phase formerly below)
Queue space 80 K		(incl. ctrl blocks & message buffers) (dynamically extendable, 4K increments)

„ Console Router Trace available

Only for diagnostic purposes

CORCMD command with options

TRACE	returns current trace setting
TRACE=ON!OFF	sets console router trace on!off
TRACE=n	changes the size of the trace area to n KB (Default is 8K, in 31-bit System GETVIS)
TRACE=END	returns the trace storage to the system

DEBUG command also sets the console router trace on!off

Therefore, CORCMD is only required if a console trace is needed during IPL.

For more information on VSE/ESA 2.1 Console Support refer to the VSE/ESA 2.1 documentation, or to the following survey article:

'VSE/ESA 2.1 Console Support', Summary article by Manfred Baudisch, VSE/ESA Software Newsletter, 3Q/4Q95, pp20-33

As VSENEW11 PACKAGE on the IBMVSE tools disk, and on Internet as Postscript.
Was distributed 12/95 to every licensed VSE customer.

REXX/VSE Console Automation

REXX/VSE Overview

- VSE/ESA 1.3.2 optional product, in VSE/ESA 2.1 Base
- VSE Interpreter for REXX
- More flexible JCL, POWER command environment ...
- Console Automation in VSE/ESA 2.1.1

  PTF UN80995 for APAR PN71905
- CPU Monitor in VSE/ESA 2.2 Base

  Described in the 'IBM VSE/ESA Hints for Performance Activities' document

REXX Console Automation

- Easy automation of console operations
  - May help e.g. to increase batch throughput
- REXX programs

  issue console messages

  - AR commands
  - POWER, ICCF, CICS, VTAM SQL ... commands

  retrieve responses

  may invoke e.g. the REXX/VSE CPU Monitor

  Samples in 'Console Application Framework'

REXX/VSE Performance PTFs

REXX Performance PTFs

„ Assignments in REXX programs

A PTF saves VSE SVCs for value assignments in REXX for VSE/ESA V2:

APAR PN91777 PTF UN98614

Depending on the REXX program it may significantly reduce CPU-time, which -of interest for the Turbo Dispatcher- was non-parallel.

„ FINDMSG CPU-time reduction

The FINDMSG function

- allows to monitor console messages and initiate corresponding activities
- is heavily used in the REXX Console Application Framework REXXCO
- is now implemented in a much less CPU consuming way

Install the PTFs:

VSE/ESA 2.2:	APAR (PQ11868	PTF UQ13219	superseded)
	PQ18360	UQ20922	
VSE/ESA 2.3:	APAR (PQ11893	PTF UQ13366	superseded)
	PQ18399	UQ20923	

References

- REXXDOCS PACKAGE on IBMVSE tools disk and on INTERNET

- 'REXX/VSE Reference, SC33-6642-01 (12/96)

- 'REXX Console Automation on VSE/ESA 2.1'
  VSE/ESA Software Newsletter 3Q/4Q95, pp 34-40

- The REXX/VSE homepage
  http://www.s390.ibm.com/products/vse/rexx/rexxhome.html

Page Manager Enhancements -NOPDS-

NOPDS Option in Supervisor Parameters Command

- **Applicability**
  - **Only for cases where VSE paging is not possible/required:**

Real_storage > reqd_VSIZE +VIO +pg_mgr_real_space

Real storage may be 'simulated' by VM or 'real'

  - **Especially for V=V guests under VM/ESA, if DEF STOR is big enough**

Í No VSE PDS required on DASD for these cases

Native VSE or V=R/F VM/VSE guests apply only in rare exceptions to that condition, if at all

- **Rationale**
  - NOPDS was done
    - to get rid of an unused VSE V=V PDS in case all paging should/could be done by VM
    - NOT a step to leave the valuable and required concept of 'virtual storage'

- **Functionality**
  - **Naturally, all VSE/ESA functions are fully available with NOPDS**

NOPDS Option (cont'd)

- **Performance**
  - **Still VSE runs virtual**

    DAT ON is used, but no paging (page-I/Os)
  - **No CPU-time benefit, page tables and DAT still required**

    Same V=V guest/native ITR ratio as w/o NOPDS,
    if there was no VSE paging.

    Actual (native) runs showed <0.5% delta, caused by faster ALLBOUND

Hint: Do not define DEF STOR much larger than required,
it makes the working set of VSE appear larger to VM

Granularity for Page Mgr owned Address Spaces

- **Size increment refined from 1M to 64K**

  These spaces are used for page management tables.

  Used private space pages are PFIXed

  Savings of VSIZE, which has specific importance for 'NOPDS',
  saving
  real storage
  or
  DEF STOR size

RELPAG Macro Changes

„ Clear a page (discard old content).

Used e.g. by CICS at any FREEMAIN, and during program compression.
RELPAG internally uses SVC121 (X'79') with RELPAG function code '03', formerly SVC85

„ If page is not addressable (=disconnected, =not in real storage), 'no-valid-copy-on-PDS' is set to avoid a later page-in

„ If page is addressable, the following activities are done:

RELPAG activity	VSE/ESA 1.3	VSE/ESA 2.1
Clear page to 0s	No	X
Disconnect page (make page no more addressable)	X	No stays addressable
Indicate 'No copy on PDS'	X	X
At first reference of empty page in any case	PF	PF only if disconnected meanwhile
Performance effects:		
-> Nonpaging environment	Many PFs for disconnected pages	Less PFs handled
-> Paging environment	1xClear	2xClear if disconnected before use

„ No change in any physical page-I/O

Cleared page has reference and change bits off,
so a page-out is never done.
No page with 'no valid copy on PDS' is ever paged in.

„ No impact on real storage requirements (working set)

Empty page (VSE/ESA 2.1) is set at the top of the page selection queue, making the frame easily available and protecting others

More ECKD Support

More (native) ECKD Support

FETCH/LOAD for phases in libraries on ECKD devices

HardCopy File on ECKD devices (WRITE, but also READ)

Also higher block size used (4K vs 2K) for CKD and ECKD

For FBA, blocksize not changed (0.5K)

Benefits

- Savings of several hundreds of instructions per converted I/O

  (No CKD-ECKD channel pgm conversion required by supervisor)

- Reduction of HCF-I/Os up to 50%, faster PRINTLOGs and Scrolls

í **Minor component performance improvements, smaller impact on overall load**

Smarter CKD/ECKD Conversion Routine

Conversion enhancement

- 'Non-translatable' CCWs at end of CKD channel programs
  do not break already converted CCWs
  e.g. NOP, or READ CCWs outside of LR domain

- Has been retrofitted meanwhile to VSE/ESA 1.2/1.3:
  APAR DY43312, with PTF UD49232,3,4, and 7

- Also beneficial for RAMAC Array DASD/Subsystem

- Consider also APAR DY43585 (PTF UD49565,6)
  for CKD programs with multiple SEEKs

Misc. Tracing Enhancements

„ Partition Trace (PTRACE) for tracing VSE usercode

- Tracing overhead much smaller than SDAID trace
  (PER/370 only in effect if traced partition dispatched)

  - by tracing only individual partitions
  - by tracing only user code

„ LIBR Trace can be restricted to an individual partition

via a new TEST TRACE operand 'PARtition'

- Tracing overhead much smaller

VSE/VSAM Extended User Buffering

Available since VSE/ESA 2.1.2, via PTF.

For more info refer to SC33-6629-00 'VSE/ESA Enhancements', p.61ff

„ Normal User Buffering (NBF)

Buffers for VSAM ESDS CIs are located in virtual storage NOT controlled by VSAM

Applies to database systems, e.g. DL/I and SQL/DS

VSAM ACB specifies : MACRF=(CNV,UBF,MVE,...)
VSAM RPL identifies: - buffer address (AREA= parameter)
- RBA of a CI (ARG= parameter)

- **Each single VSAM request (for a CI) results in an immediate I/O**

„ Extended User Buffering (XBF)

Handles, if possible, related requests in a single I/O:

Requests in same VSAM CA will usually be processed in 1 I/O.

New parameters OPTCD=(...,NBF,...) Normal User Buffering
OPTCD=(...,XBF,...) Extended User Buffering

also in other VSAM macros (GENCB, MODCB, TESTCB)

„ XBF used by ...

ADSM/VSE: PTF UN23606 required (APAR PN76962)

DB2 5.1 : PTF UD50455/50456 required (APAR DY44537)

Misc. items

„ **SA FASTCOPY functionwise identical to standard FASTCOPY**

- OPTIMIZE parameter with Read Track for ECKD
- Results in faster SA Restore (+Backup) for ECKD

„ **IDCAMS VSAM buffer enhancements (e.g. REPRO)**

Buffers may now reside above the line

Í Bigger buffer space possible for reduced number of I/Os,

used if ample GETVIS available

Default buffer allocation is dynamic

In case DLBLs for REPRO do not contain BUFSP/BUFND/BUFNI values:

- More GETVIS-24 space is used

and, if GETVIS-31 is also available

- Even more total GETVIS is exploited

Some partition GETVIS space is left
(if other tasks may be present in same partition, seldom)

Í **Better IDCAMS default performance than before, if partition big enough**

Misc. items (cont'd)

„ DUMP buffers for dynamic partitions in dynamic space GETVIS

(IDUMP, SDUMP, ABEND Dump)

-> Allows concurrent dumps for dynamic partitions
even when System GETVIS-24 is currently exhausted

Concurrent DUMPs	VSE/ESA 1.3	VSE/ESA 2.1
Static partitions	YES *	YES *
	NO	NO, but faster
Dynamic partitions	YES *	YES, dyn. space
	NO	GETVIS used

- No concurrent DUMPs in VSE before VSE/ESA 1.3

- In VSE/ESA 1.3 50K SVA-24 GETVIS were requested first.
If not available, a (single) 5K area in $IJBSDMP was used,
serializing all dumps

* If 50K SVA-24 GETVIS request successful

Í DUMP speed may improve in certain cases

by requesting as much GETVIS-24 as possible
(no more 5K as 2nd choice, also 80K max)

- Dynamic partitions: GETVIS from dynamic space
- Static partitions: GETVIS from SVA-24

Internal DEBUG for Trouble Shooting

DEBUG Facility Impact on Performance

Refer e.g. to the 'Hints and Tips for VSE/ESA' booklet.

Ù **Background**

All VSE supervisors are capable of 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:

- DEBUG	to display DEBUG status (default is OFF, except at IPL)
- DEBUG ON (,nnnK)	to switch it on (system-wide)
- DEBUG part-id	to switch it on for a partition only
- DEBUG SHOW(,CUU=cuu)	to display current buffer contents (PSHOW = previous, NSHOW = next buffer)
- DEBUG OFF	to switch all DEBUG traces OFF
- DEBUG END	to free DEBUG areas in System GETVIS-24

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

DEBUG areas:

a) Statically in the dynamic part of supervisor:

DEBUG=NO: 1x4K area (new, for DEBUG during IPL, reused)

b) Dynamically in SVA-31

DEBUG ON: being set for a DEBUG=NO supervisor (3x16K)

í **DEBUG=YES is an internal supvr option only, NEVER to be used by customers**

Internal DEBUG for Trouble Shooting ...

DEBUG Impact by increased CPU-time

Sample overhead results (DEBUG ON vs OFF)					
Base CPU utilization	26%	PACEX1 %	PACEX4 %	RAMP-C %	DSW %
CPU-time /job or tx	+23%	+ %	+ %	+ %	
(variation range)	(8%-33%)	- -	- -	- -	
#instr. total	+25%	+ %	+ %	+ %	
#instr. in SUPVR stat.	+70%	+ %	+ %	+ %	
Batch ET, tx RT	+ 5%	+ %	+ %	+ %	
(variation range)	(0%-10%)	- (10%-40%)	-		

- Measured results for VSE/ESA 2.1.0 on 9221-170, with cached 9345 DASDs (e=estimated)
- Values apply to shipped VSE/ESA 2.1 supervisors with standard dispatcher.
- CPU-time overhead depends on 'relative intensity of traced supervisor events'
- Elapsed or Response time overhead depend on CPU-time overhead and CPU-utilization

DEBUG ON in VSE/ESA 2.1 vs VSE/ESA 1.3:

DEBUG ON overhead increased vs VSE/ESA 1.3
by more debug entries for better serviceability

- **CPU-time overhead in debug situations or for test systems acceptable (only then)**

- **DEBUG ON in a Turbo Dispatcher Environment**

  - TD DEBUG controlled via 'standard' DEBUG
  - Separate TD DEBUG area (1x32K) in PFIXed SVA-31 space
  - Similar relative overhead measured as for 'standard' dispatcher: e.g. +26% CPU-time/tx on a 9121-480
  - Higher elapsed time overhead for 'standard' DEBUG by serializing effect of 'standard' DEBUG entries

VSE/ESA 2.x IPL Procedures

Ù Shipped IPL procedure ($IPLESA.PROC)

- „ must (function-wise) work in all environments
- „ must be modified for optimal performance

  Depending on workload environment and on space requirements
- „ contains performance relevant parameters

Excerpt from $IPLESA.PROC

```
$$A$SUPX,VSIZE=180M,VIO=512K,VPOOL=64K      *1)
ADD  FDF,FBAV        *2)
SYS  JA=YES,BUFSIZE=1500,NPARTS=24,SPSIZE=0K,PASIZE=30M   *3)
DPD  VOLID=DOSRES,....
DPD  VOLID=SYSWK1,....
DPD  VOLID=DOSRES,....  *4)
DPD  VOLID=DOSRES,....  *4)
SVA  SDL=300,GETVIS=(768K,3072K),PSIZE=(256K,2000K)        *5)
```

*1) 180M is the VSIZE after Base Install (was 120M in 2.1).
Is 250M in 2.4.

*2) Label Area on Virtual Disk is default for VSE/ESA 2.4

*3) Default CHANQ in most cases sufficient.
BUFSIZE=1500 also. Check both via SIR.
No FASTTR in VSE/ESA 2.4; check BUFSIZE, may be reduced.
NPARTS=44 in VSE/ESA 2.4.
SDSIZE=64K is default, required e.g. for SDAID

*4) In case you still should have VSE paging in spite of ESA
(but don't 'DIM' to exchange file I/O with page-I/O),
move these DPD extents to other volumes

*5) SDL increased to 700 for VSE/ESA 2.4.
Not all SVA-24 eligible modules (e.g. from vendors) are
desirable to reside in SVA-24.

GETVIS-31 increased to 6M in VSE/ESA 2.4.
PSIZE-31 increased to 3M/6M in VSE/ESA 2.3/2.4.

VSE/ESA 2.x IPL Procedures (cont'd)

- **Required/Suggested modifications to IPL proc**
  - **In any case, specify ample GETVIS-31**

    Use e.g. 6M to start, reduce later if you like
  - **PSIZE-31 should be ample**

    Avoid that any SVA-31 eligible phase is loaded below.
    Start e.g. with 8M.

- **Regarding virtual storage tuning (including SVA-24 (VLA-24) optimization) see section 'Space Optimization'.**

- **Regarding data space related values see section 'ACF/VTAM 4.2'**

TPBAL ('Telecommmunication Balancing') Function

„ Background

A function designed for heavy paging systems, still included in VSE/ESA

„ Function

Temporarily suspend page-fault handling for n low priority batch partitions

„ Notes

1. An individual batch partition is affected only if

   A TPBAL value (n>0) was set to include this partition (all eligible partitions and dynamic classes are being displayed when TPBAL is set)

   A TPIN macro (SVC'58') was issued and not yet reset by a TPOUT (SVC'59') from any partition (TPIN and TPOUT are usually only isssued by CICS and/or VTAM)

   In the TPBALed batch partition a page fault occurs in user code

   A STATUS command snapshot may show 'page fault bound' for the TPBALed partition

2. The TPIN and TPOUT macros only cause SVCs, if TPBAL is active (n>0)

TPBAL Function (cont'd)

„ Assessment

Non-paging VSE

1. For a non-paging system TPBAL is without effect/benefit, but has the following side effects:

   - additional SVCs cause additional dispatcher calls,

   - additional non-parallel transitions (for Turbo Disp.)

Paging VSE

1. Since TPIN is only valid for small time intervals, the effect is very limited

2. This function does NOT orientate itself at average paging rates

Í TPBAL is NOT a dynamic function nor a permanent one

„ Recommendations

Í Usage of TPBAL is NOT recommended

Leave it off (n=0) as is the default and the CICS recommendation

For Turbo Dispatcher, side effects are more harmful

„ Paging-rate dependent 'VSE load levelling'

Activation/deactivation of low priority batch partitions
is done by VSE/AF based on processor dependent fixed values,
not alterable by user.

Í VSE Load levelling does not consider individual paging situations
e.g. #paging devices, device speed, ...

Í For CICS production with concurrent Batch, provide sufficient real storage (the only choice)

Dynamic vs Static VSE Partitions -Performance View-

- **Functional Aspects**

10 dynamic classes with 'many' dyn. part.

vs 12 static partitions in total

Setup in dynamic class table(s) DTR$DYNn.Z

controlled by POWER

Few services not allowed

(POWER), BTAM and VTAM must run in static partitions, and no SYSFIL support
(PFIX available)

'Forced cleanup' at EOJ

no remaining 'side-effects'

- **Performance Aspects**

Partition Balancing within each dynamic class

With TD, any dynamic PARTITION (not CLASS) gets same time slice as a static one

VSIZE for a dynamic partition only req'd when partition (POWER job) active

Dynamic Space GETVIS available

Is a specific GETVIS request (SPACE=YES means storage to be kept across job steps, but no sharing required).
When used in a dyn. partition, avoids SVA-24 GETVIS usage: e.g. DUMP, LIBR, Job Control, Vendor programs (key 0 req'd)

Small overhead for dynamic partition (=POWER job) startup

Refer to next chart

Dynamic Partition Overhead

- Small overhead at initialization and deactivation
  - No extra overhead after partition is started
- Overhead varies with JCL setup
- Some overhead measurement results

Deltas	Minimum	Default	Delta LIBDEFs	Delta ASSGNs	
CPU-time (sec) 0.1	0.3	0.1	0.1		
# IOs	26	53	22	5	

- CPU-time for 9221-150
- VSE/ESA V1 with NOLOG, newer results under way
  (VSE/ESA V2 with slightly increased overhead)

- Minimum refers to a small JCL
- Default refers to the shipped JCL, with LIBDEF proc
- LIBDEF proc used (3 LIBDEFs, 8 sublibs each)
- ASSGNs with 20 logical assignments in total

-> Elapsed Time overhead less than about 1 sec

- Dynamic partitions are very good for
  - **POWER jobs that are not too short**
  - **CICS partitions**

    May save about 100K per CICS partition via Dynamic Space GETVIS
  - **Data base partitions**

    CA Datacom DB e.g. saves supervisor services if run in dyn. partitions (they all have same storage key, so no key 0 req'd).
  - **other long running applications or subsystems**

Some APARs/PTFs of performance interest

* 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).

This PTF holds for VSE/ESA V1 only, Version 2.1 has this improvement incorporated w/o PTF.

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

* DY43528 UD49442 VTAM GETVIS performance improvement

With this PTF, VTAM 4.2 will use GETVIS subpools, in order to quickly find appropriate GETVIS storage for network activation. This PTF is included in VSE/ESA 2.1 as of 05/19/95.

* PN70494 UN76442 CICS/VSE 2.3 MRO performance
UN76442

This PTF allows to specify a new parameter (NOFS) in the SIT option (ISC), giving CPU-time improvements for all those cases where ISC is required, but no Function Shipping is used (refer to CICS 2.3 part of this document).

* VM59317 UM27166 VM CCW Fast Path performance

With this VM/ESA 1.2.2 PTF, it is avoided that all VSE channel programs with bits set for record caching and/or regular data format are aborting the VM Fast CCW translation.

* DY43585 UD49565 Misc. problems, incl. performance problem
UD49566 for specific CKD channel programs on ECKD

With this PTF also a local performance problem is solved, which resulted from CKD to ECKD converted channel programs, containing multiple SEEKs. Now, unit checks are avoided which caused redrives of the original CKD channel program.

* DY43684 UD49610 Misc. problems, incl. performance problem
UD49612 for GETVIS
UD49613 (ESA 2.1.1 refresh, icl. TD perf. impr.)

With this PTF a GETVIS pathlength reduction is provided for several cases, taken over from VSE/ESA 1.3.

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

* DY43770 UD49684 VTAM 4.2 patch to reduce SVA-24 usage

  This PTF solves the problem that the initial set of VTAM buffers was created in SVA-24 (in spite of SVA-31 eligibility).
  Occurred when ISTORCBP phase was not loaded above the line, since the VTAM partition was totally below the line.

* DY43652 UD49617 New VSAM 'performance function' for ESDS
  UD49618 files with CI-mode

  With this PTF, VSAM provides an extended buffering (RPL OPTCD=XBF), which allows several ESDS CIs to be chained in a single I/O.
  This function is currently used by ADSM.

* PN76962 UN83606 Performance enhancement for ADSM to exploit VSAM Extended User Buffering (see last PTF)

  ADSM exploits this new VSAM function to reduce the number of VSAM I/Os for its ESDS files.
  SIZE=3800K is required for the ADSM partition.

* DY43697/8 UD49662 Some functional and performance enhancements:
  UD49664 Turbo Dispatcher improvements,
  CKD/ECKD conversion routine enhancements,
  CACHE SUBSYS=cuu,REPORT provides summary data

  With this PTF, e.g. the CKD/ECKD conversion routine is smarter to CKD programs with format writes if no sector value is given.
  Also, for native VSE, all data of all devices at a subsystem are now accumulated to directly provide the overall hit ratio.

* DY43844 UD49764 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.

VSE/ESA 2.1/2.2 Performance PTFs ...

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

* DY43952 UD49914 VSAM performance PTF

This PTF improves VSAM data compression with ICCF started.
It also reduces VSAM CPU-time by avoiding SECTVAL SVCs
for SD and TD, provided the file is NOT defined with IMBED,
and provided the partition crosses the 16M line (the latter
condition was used so far for compatibility reasons, though
only 1K per file is used for that. It was removed in a later
step).
It requires a TD PTF (for APAR DY43919) and is not contained
in VSE/ESA 2.1.2, but standard in VSE/ESA 2.1.3.
Please make sure that the performance benefit of this PTF is
not reduced by a newer VSAM PTF, i.e. make sure that also VSAM
PTF UD50015 is applied.

* DY44070 UD49933 VSAM catalog mgmnt, VSAM managed files 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).

* PN84947 UN92489 VSAM performance degradation in COBOL/VSE
* PN84949 UN92490 VSAM performance degradation in COBOL/II

This PTF solves a performance problem occuring for VSAM MODCBs/
SHOWCBs to modify/extract record length before and after
PUT/GET. These specific macro usages now expand to the same
effective code as for COBOL/VS.
(Additional MCs and dispatcher calls saved for VSE/ESA 2.1)

* PN91777 UN98614 REXX GETMSG fct and general performance PTF

This PTF corrects a GETMSG problem and reduces REXX CPU-time
for programs with many REXX assignments.

Re-compile required for Batch programs to benefit from better
macro extensions]

* DY44311 UD50183 Misc patches plus GETVIS command enhance-
UD50187 ments
UD50188

This PTF (besides providing the Turbo Dispatcher version code)
includes 2 additional options of the GETVIS SVA command
(regarding subpool info and details)

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

* DY44442 UD50251/52 Misc. plus SIR SMF,cuu command

  This PTF also includes a supervisor PTF for parallel POWER and an enhanced GETVIS SVA,DETAIL display

VSE/ESA 2.2.1 is available since 97-04-25. It contains all base and optional product PTFs with cutoff date until 97-03-06.

* DY44776 UD50614 VSAM SECTVAL avoidance, extended

  This PTF is an extension to DY43952. It avoids VSAM SECTVAL SVCs also in cases, where no GETVIS-31 actually is available.

  Partition GETVIS-ANY requirements increase by
  - 384 bytes per ESDS or RRDS file
  - 768 bytes per KSDS or VRDS file

  With this PTF, only the following conditions must be fulfilled on order to benefit from VSAM SECTVAL avoidance:

  - Running under VSE (not as CMS VSAM)
  - Not an FBA device
  - File not defined with IMBED (i.e. REPLICATEd seq. set)
  - Program not running in ICCF I/A partition

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.

Background Info

Ù The following files are sharable (across VSEs)

DLF (Lockfile, must be shared)
POWER spool
VSE/AF libraries, in general
IJSYSRS
PRD1, PRD2 (but only if also VSAM catalog shared)
VSE Control File
VSE History File
VSE/VSAM catalogs, spaces, files ...

Ù The following files are NOT sharable (across VSEs)

BAM files
Label file
DTSFILE
...
Hardcopy File
Page Data Set
Recorder File

Ù Data Integrity is key

Update integrity: Ensure that no concurrent update is done ('last wins')

Read integrity: Ensure the most recent update of a record is read (and not an elder image from old place)

Ù The VSE Lock Manager controls all lock requests:

Locks are handled/made INTERNAL, if possible, or EXTERNAL, if required.

The Lock Manager always uses EXTERNAL locks, if the VOLID parameter of the DTL (Define The Lock) belongs to a device ADDed with ,SHR.

EXTERNAL locks always mean Lockfile accesses (I/Os).

The Lockfile is defined in the DLF IPL command(s).

Cont'd

More Background Info

Ù Lockfile Access

The following CCWs are used, to coordinate the accesses from several VSE systems to the same Lockfile:

RESERVE CCWs

to reserve a whole volume (deny all accesses from other VSEs) until the volume is released with ...

RELEASE CCWs

Ù Locks in VSE Lockfile

For cross-system shared resources locks are being set/released in the VSE Lockfile.

The unit of locking may be a record, a file, a volume or any other S/W resource.

The VSE Lock Manager locks a RESOURCE NAME and thus does not know what is hiding behind that name, i.e. he cannot correlate a lock to a file or volume.

The VSE Lock Manager does NOT know, whether a resource is actually cross-system shared or not.

Only that the resource is on a volume ADDed with ,SHR

... and thus might be shared across VSE systems.

DASD Sharing Overhead

General DASD Sharing (Locking) Overhead

Ù Additional RESERVEs to the Lockfile volume

In order to coordinate the accesses from several VSE systems to the common Lockfile ...

RESERVE is used, which reserves the whole volume and does NOT allow anybody to access this volume until it has been RELEASEd.

Ù Additional I/Os to the Lockfile to read and update the lock record

Each lock is entered into the Lockfile and valid for the defined resource until reset.

Ù Additional RELEASEs to the Lockfile volume

A RELEASE is done each time after a lock has been set or reset in the VSE Lockfile.

Ù Unsuccessful trials to access any file on the (always shared) Lockfile volume

When the total volume is still RESERVEd, a 'Device Busy' occurs. VSE waits 1 sec (thus introducing also a time delay) before a Lockfile access and/or to get a Lock is retried.

... on top of usual device contention

For any file on a shared DASD volume always the usual device contention applies (#IOs/sec per actuator), as for non-shared volumes.

More Details on Lockfile Accesses

Ù **Principal Scheme for Lockfile Accesses**

*** SVC110 (LOCK)**

```
          RESERVE
1)    READ  resource  status
------
2)    WRITE  lock  (lock  resource)
          RELEASE
```

=== Hold the LOCKed resource ===

*** SVC110 (UNLOCK)**

```
          RESERVE
3)    READ  same  lockfile  block  (might  have  changed)
------
4)    WRITE  lock  (unlock  resource)
          RELEASE
```

- Each READ or WRITE applies to a 512 byte block, where also other resources are contained

- RESERVE is always chained in the same channnel program, RELEASE is only chained if possible

- If a resource is already locked ('failed'), a separate RELEASE of the device is done, and then the LOCK SVC is re-issued later for the next trial.

- If an external LOCK is immediately sucessful, in total 2 READs and 2 WRITEs are done (besides RESERVE/RELEASEs) and in total 2 SVC110s.

DASD Sharing Overhead (File Type specific)

Ù BAM files

(not sharable per se)

„ Do not have the general locking overhead above, even if the volume it is on is shared or not.

Ù VSE/POWER Queue and Account file

„ Are shared via the above mechanism to the Lockfile

„ LOCK macros to the VSE Lock Manager are used on a workload driven time sliced basis

„ Suffer from the above locking overhead,

but only when the pertinent file(s) have been defined at POWER generation as shared.

(Queue and Data File combined, with or w/o the Account File).

Ù VSE/AF libraries

„ Have this locking overhead through the VSE Lockfile, if the first (or only) extent is on a shared DASD:

- Locking for exclusive WRITEs for sublib-index, bitmap (no more at OPEN/CLOSE as for the old VSE libraries)

Ù VSE/VSAM catalogs and files

Refer to next charts

Some VSAM Background Info

Ù The following resources can be cross-system shared

- Volumes (via ',SHR' in ADD statement of all VSEs)
- Catalogs (via residing on a shared volume)
- VSAM space (" " " " " ")
- VSAM files (via VSAM SHROPT 4 4)

Ù In order to cross-system share a VSAM file

- the catalog must be on a shared volume
- the file (VSAM space(s)) must also reside on shared volume

Ù Any shared VSAM file causes LOCKs (even if shared only within the same VSE):

VSAM SHR Option	OPEN, CLOSE, file extens. NOUPD	READ UPD	READ	WRITE	
SHR 1 or 2	X	-	-	-	
SHR 3 Don't use	X	-	-	-	
SHR 4 (=4,3)	X	-	x	x	
SHR 4 4 (=4,4)	X	-	X	X	

X means EXTERNAL LOCKs
x means INTERNAL LOCKs (EXTERNAL if on shared volume)

VSAM issues the LOCK macro, when a lock may be potentially required.
In nearly all cases, VSAM sets the External Flag in the DTL, in order to allow the Lock Manager to do EXTERNAL locking via the Lockfile.

Ù If the catalog is on a shared volume ... any LOCK to a VSAM file (even if nonshared) results in an EXTERNAL LOCK

This also applies to OPENs of 'non-4 4' SHR option files, in order to prevent other VSEs from concurrent access

Recommendations for DASD Sharing

- **Lockfile definition**

  - **Do not specify much more CPUs than needed in the DLF NCPU parameter (2-31)**

    Speeds up Lockfile I/Os

- **Lockfile location**

  - **Put the VSE Lockfile, if possible,**
    - on a separate volume (native VSE)

      Use a fast READ/WRITE-cached one.
      Any other file will suffer from RESERVEs

    - on a VM Virtual Disk (VM/VSE)

  - **Put non-shared data on non-shared volumes**

    This is a general rule which does not only bring performance benefits, but also is reasonable for non-performance reasons.

    - High benefit for
      - VSAM catalogs and files
      - AF libraries
      - SQL/DS databases

    - No benefit expected by moving to a non-shared volume for
      - BAM files
      - POWER files

  - **Separate shared from non-shared data on different sets of volumes, whenever possible.**

    - Reduces the LOCK overhead (less Lockfile I/Os)

    - Reduces the problem of not being allowed to have 2 volumes with same VOLID on 1 VSE system

Recommendations for DASD Sharing (cont'd)

- **Avoid, whenever possible, to ADD volumes as ,SHR**

  - Reduces DASD sharing overhead, for non-shared files

- **Use separate VSAM catalogs for cross-system shared data and for non-shared data**

  - Avoids the LOCK overhead (Lockfile I/Os) at OPEN time for 'non-4 4' SHR option files.

- **In critical cases ...**

  Look at lock statistics of the SIR display

```
LOCKS  EXT.=        450        LOCKS  INT.=      3636    !  All  lock  trials
       FAIL  =          6                   FAIL  =       185    !  Failed  lock  trials

LOCK  I/O    =     1820        LOCK  WRITE=       900    !  Lockfile  I/Os
```

Requests for EXTERNAL locks may be shown, even when no Lockfile has been defined, i.e. when no DASD Sharing is being used. If many EXTERNAL lock trials fail (>10%), and no such resource is scarce, check the size of your Lockfile in the DLF command.

LOCK I/O here refers to all LOCK READs and WRITEs, not counting potential separate I/Os for RESERVE or RELEASE.

Refer to the SIR description in 'Hints and Tips for VSE/ESA'.

You may also look at SVC110 (X'6E'=LOCK/UNLOCK) frequency of the Turbo Dispatcher SIR MON SVC count display.

Also, if even more details would be required, ...

refer to the VSE/ESA 2.4 Supervisor Diagnostic Reference to exploit even more available lock counters.

Recommendations for DASD Sharing ...

Sharing Data between different VSE releases

- **Background Information**

  The format of the VSE Lockfile (and the lock records) has NOT been changed

  at least since VSE/SP 4.1.

  NCPU values between 2 and 31 (default 4) were available in all releases since VSE/SP 4.1

  Every change of the NCPU value needs a new DLF command and thus a new formatting of the Lockfile

- **Conclusions (VSE/SP 4.1 and up)**

  - **It is irrelevant which VSE releases participate in DASD Sharing**
  - **Any VSE release can be used to format the Lockfile, it need NOT be the newest VSE release participating**

    Naturally, doing so is a good practice

More Info

- VSE/ESA Guide to System Functions SC33-6611,
  Chapter 'DASD Sharing with Multiple VSE Systems'

- VSE/ESA Hints and Tips brochure, 12/97,
  'Cross-Systems Sharing', page 156

- VSE/VSAM Performance and Tuning, by Dan Janda, WAVV 09/98,
  Part D 'Data Set Sharing'

PART E.

VSE/POWER

The following VSE/POWER levels are applicable:

6.1.0 (VSE/ESA 2.1.0 and up)
6.1.1 (VSE/ESA 2.1.2 and up)
6.1.2 (VSE/ESA 2.2.0 and up)
6.3.0 (VSE/ESA 2.3.0 and up)
6.4.0 (VSE/ESA 2.4) refer to 2.4 part

POWER release numbers e.g. from 'D STATUS'.

„ 3800 Printer Support

„ POWER Enhancements in subsequent releases

Separate chart

„ POWER Virtual Storage

„ More POWER Hints

- Physical printing
- Tracing
- Accounting
- Partition priority
- Shared Spooling and PNET

3800 Support

„ Use the 3800 Support (was IBM Program 5747-CC1)

Now fully integrated into VSE/ESA 2.1.1 Central Functions.
Was an ICR program, obsoleted as such in 94

3800 IOCS modules adapted and standard since VSE/ESA 1.3

Refer to 'Installation Info' on next foil

„ Spooling requests are buffered in VSE/AF, before POWER involved.

Also applicable if real printer is not a 3800

„ Performance benefits

Saves CPU-time by not handling POWER print lines individually

í Between 3% and 50% CPU-time reduction:

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

Base is VSE/ESA 2.1 default POWER setup (DBLK=4K, ADD FEE=PRT1)

„ More info

- 'Boosting VSE Performance' by T.Pylant,
  Enterprise Systems Journal 10/93, pp 74-77
- Dan Janda's 3800 IEBIMAGE writeup
  http://www.s390.ibm.com/products/vse/vsehtmls/3800icr.htm
- the following pages

3800 Support, Notes

„ FCBs

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

1 via IEBIMAGE (was 5745-SC-IMP and part of 5747-CC1) for spooling (define it 1 inch larger than actual paper size)

1 via Assembler for printing on the (real) PRT1

„ Console message PTFs

To avoid frequent information messages 1Q4xI on the console, be sure to have incorporated

APAR DY43856 PTF UD49806 (VSE/ESA 1.3/1.4)
APAR DY43978 PTF UD49802 (VSE/ESA 2.1)

With these PTFs, these messages are only shown if data have not been printed.

New enhancement PTF:

APAR DY44335 PTF UD50262 UD50263 (VSE/ESA V2)

Currently, at each start of a print job, message 1Q41I appears 'WRONG PRINTER/PUNCH FOR...'.
This PTF anhancement allows a POWER startup option 'SET 1Q41I=NO' to suppress this message.

„ Installation info

	3800 IOCS modules (3800-PRT)	IEBIMAGE for FCBs (3800-LIB)
ICR 5747-CC1 (obsolete)	contained, not adapted to ESA	contained
VSE/ESA 1.3/1.4 in VSE 2.1.0	included and adapted to ESA	not included
VSE/ESA 2.1.x (x>0) in VSE	included and adapted to ESA	included

Using the POWER 3800 Printer Support ...

Definitions required

- **Define your DUMMY printers as 3800**

```
ADD   F20:F2F,3800
...

Do  this  also  under  VM,  to  avoid  that  by  sensing  VSE  resets
this  device  type  to  whatever  type  VM  returns
```

- **Specify them as the printers for your partition**

```
PSTART   PRINTERS=   list
PSTART   BG...
READER=cuu
PRINTERS=F20,F21,...  (as  printers  are  needed)
PUNCHES=...
...
```

- **Assign your printer logical units to these 3800 dummy printers during partition startup**

```
ASSGN   SYSLST,F20
ASSGN   SYS006,F20
ASSGN   SYS020,F21
...
```

- **ADD and PSTART real printers with their real characteristics and addresses**

```
ADD   10E,PRT1     (or  whatever)
...
PSTART   LST,10E,abc
       or
PSTART   LST,00E,abc,,VM     (if  spooled  to  VM)
```

FCB Considerations for 3800/PRT1

„ 3800 page layout

A real 3800 cannot print on the first and last .5 inch of a page

Í Actual printable size is physical page size minus 1 inch

To obtain same page layout as before on PRT1 ...

„ Default 3800 FCB must be created (via IEBIMAGE) 1 inch larger than default PRT1 FCB

Also applies to each FCB in the FCB=fcbname operand of the * $$ LST card

Use the SETDF AR-Command to specify a default FCB for the 3800:
SETDF cuu,FCB=3800 (If phasename is FCB13800, as used below)

„ Example for a 3800 FCB

Fits to a PRT1 FCB with 8 LPI and a FORMS length of 8 inch:

```
* $$ JOB JNM=FCB13800,CLASS=A,DISP=D,PRI=3
* $$ LST DISP=D,CLASS=H,DEST=(,HARTMANN)
// JOB FCB13800
// LIBDEF PHASE,CATALOG=IJSYSRS.SYSLIB
// OPTION CATAL,NODECK
* THIS JOB CREATES AN FCB FOR 9 INCH PAPER ON A 3800
*   CHANNEL 1   ON LINE 5
*   CHANNEL 12  ON LINE 56
*   PRINTABLE FORMS LENGTH IS 8 LPI X 8 INCH = 64 LINES
*   CATALOGED PHASE NAME IS FCB13800
*   IT CORRESPONDS TO A PRT1 FCB WITH 8 INCH FORM LENGTH
// PAUSE
// EXEC IEBIMAGE,SIZE=AUTO
      FCB    LPI=8,
               CH1=5,
               CH12=56,
               SIZE=90                    - in tenth of an inch = 9 inch
      NAME 3800,REPLACE=YES - resulting phase name is FCB13800
/*
// EXEC LNKEDT,PARM='MSHP'
/*
/&
* $$ EOJ
```

Note: The FCB's must be cataloged to IJSYSRS.SYSLIB,
otherwise SETDF is not able to load the FCB

VSE/POWER Performance Enhancements

- **POWER 6.1.0 (VSE/ESA 2.1.0)**
  - Max. DBLK size increased to 64K
  - Increased Output Buffer size
  - Increased max. PNET Buffer size
  - Enhanced Viewing of Queue-Entries
  - Parallel Browse of Queue-Entries
  - New Output Segmentation Macro

- **POWER 6.1.1 (VSE/ESA 2.1.1)**
  - 3800 Printer Support fully integrated

- **POWER 6.1.2 (VSE/ESA 2.2.0)**
  - More parallel POWER for Turbo Dispatcher

    Refer to TD document

- **POWER 6.3.0 (VSE/ESA 2.3.0)**
  - Bigger DBLK default sizes
  - Bigger PNET default buffer size
  - POFFLOAD PICKUP

- **POWER 6.4.0 (VSE/ESA 2.4)**
  - POWER Queue File above the line

    Also retrofitted to VSE/ESA 2.x
  - Max. SETPFIX LIMIT increased to 2M
  - Up to 32 Data File extents

POWER DBLK Improvements (2.1.0)

Ù Maximum DBLK Size Increased from 12K to 64K

(Actually 64K-512 bytes. If a specified value is too big,
POWER truncates it to the maximum value with msg 1Q3BI)

Applicable to all POWER spool file activities
(at spool and print time, incl. PSF print)

Ù Benefits:

Allows to fully exploit CKD/ECKD track capacity

(beneficial for all non-RVA I/O subsystems)

Reduces POWER data file I/Os, both at 'spool' and at 'print' time.

up to a factor of tracksize/12K

Overall noticeable for spool-intensive workloads or e.g.
print servers with optional PSF on top
(No increase of POWER data file I/Os by introduction of
backward pointers in the POWER data file).

But bigger DBLK buffers in partition-GETVIS-24,
need more space below the line.

Higher speed of POFFLOAD operations to/from tape by fewer DASD and tape I/Os

POWER DBLK Improvements (cont'd)

Ù DBLK Performance Observations

- 9221-170 with cached 9345 DASDs
- Several POWER spooling workloads
- DBLK=45040 vs DBLK=4080 (default)

Relative Spool Intensity	Workload	Elapsed time	Total CPU-time	Total I/Os	
Low		PACEX	-0.5% +-0%	-0.5%	
Medium	Your Batch				
High		SPOOLINT	-19% -7%	-80%	
SPOOLINT was a LIBR LIST of a supervisor					

Í Benefit dependent on relative spool intensity

Ù Aspects for POWER DBLK Selection

- The bigger DBLK, the lower the number of POWER Data File I/Os, but the higher is the POWER GETVIS-24 requirement.

- Good track exploitation is given at
  DBLK=tracksize and
  =tracksize/2

- Small outputs suggest lower DBLK sizes, if space is a problem

- The last DBLK of a job always is only partially filled.

- The total size of the POWER Queue File need not change, if DBLKGP is reduced

- Bigger DBLKs may force you to move the POWER Queue File above the line (VSE/ESA 2.4, retrofitted to VSE/ESA 2.x)

- You cannot select DBLK >32K

  - if POFFLOAD tapes are sent to microfiche devices
  - if PNET is used
  - if a vendor product dictates this
    (BIM products currently are limited to 48K)

POWER DBLK Improvements (cont'd)

Ù DBLK Defaults

DBLK=0 so far resulted in DBLK=4080 byte.

In VSE/ESA 2.3 the values have been optimized and adapted to the individual device type:

7548 for 3390s, 7476 for 3380s, 7548 for 9345s

Ù DBLK Recommendation

í 2 DBLKs per track is a good compromise

27998 for 3390s, 23476 for 3380s, 22928 for 9345s

together with the POWER Queue File above the line

Ù DBLKGP Selection

DASD space allocations on the POWER Data File are done in DBLK-groups:

DBLKGP DBLKs per DASD allocation unit

If DBLK is increased, DBLKGP may be reduced, in order to keep (about) the same total size (in bytes) of the POWER Data file.

BUT, this really is NOT a must.

Instead, if you leave (or even increase) DBLKGP, you may benefit especially when the DBLKGP trace is still active.

So DBLKGP should not be too low.

POWER I/Os to Disk

Summary

Ù POWER Files on Disk

POWER File	DASD Extents	Mult. I/O Blocksize	Data Distribution	Access	
Data File	X	1 DBLK *4	1 DBLKGP *1	only on 1 extent *2	Seq.within DBLKGP
Queue File	-	12K(2.4) 4K(2.3)	Multiple jobs in same block	READ(once): seq WRITE: random	
Account File	-	1 record	- *3	Purely seq.	

- VSE/POWER uses 1 CCB per disk file extent (refer to explanations, below)
- Also EXCP REAL is used (DTFPH)

*1 DBLK written to disk as soon as full
*2 Only last DBLKGP of an extent can be spanned
*3 POWER Account File uses variable records/unblocked
*4 Up to 15 extents (32 for VSE/ESA 2.4, via PTF)

Ù Type of Channel Programs: CKD (and FBA)

CKD converted to ECKD by VSE Conversion Routine,
i.e. also no caching bits used

Ù I/O Balancing

Data File data are split across Data File extents, on the base of DBLKGPs

í Some type of balancing across volumes,

when POWER Data File extents are on different volumes
(VSE/POWER does NOT consider volumes)

Cont'd

POWER I/Os to Disk ...

Ù I/O Concurrency

1 CCB per extent allows ...

„ I/O preparation asynchronous to POWER I/Os

- Move of data into pre-PFIXed I/O area
- Update CCW chain to reflect new target on DASD
  (No CCW translation required since EXCP REAL)

This benefit applies independent of volume selection

„ Overlapped I/Os

- Up to 1 concurr. disk I/O per extent

(in case all extents are on different volumes)

- Worst case: 1 POWER disk I/O at a time
  (in case all extents are on 1 volume)

Recommendations

í DBLK (and DBLKGP) selection is most important

Tracksize/2 is recommended, refer to POWER DBLK charts

í Increase number of POWER Data File extents

E.g. to 3 or 4, best on different volumes
(same device type, up to 32 allowed for VSE/ESA 2.4,
via APAR DY45190).

Needs SYS002 to SYS017 in ascending order, as volumes do
(refer to 'VSE/POWER Administration and Operation').

- Allows overlapped Data File I/Os
  in case of different volumes

POWER Data File Improvements (2.1.0)

Ù New Output Segmentation Macro IPWSEGM 'Program driven segmentation'

- Allows customer and vendor applications to segment output and to select output attributes for next segment

- Can be used instead of the 'old' SEGMENT macro, which resulted in an SVC2 to load the LTA transient $$BSGMNT. That transient resided longer in LTA, since LTA was held during I/O to POWER files. Also, as all transients, this was 'non-parallel' code for the VSE/ESA Turbo Dispatcher.

- Can be used now concurrently by >1 partition (LTA no more required)

- Also helps to avoid LTA seizing by BAM OPEN for labelled spool tape (DISP=T in POWER statement)

NOTE: The calling program can also be a 31-bit program, if -at program asssembly time- the MFG option was used. (This option allows to dynamically 'move' the CCB and CCW from the expanded macro to an area below the line)

For more info refer to the VSE/POWER Application Programming Guide (SC33-6736).

„ Benefits:

- LTA offload, since no more used

POWER Queue File Improvements (2.1.0)

Ù Enhanced Viewing of Queue-Entries

- Exploit existing forward pointers, establish new backward pointers for forward and backward scrolling

Positioning in POWER members		
	ESA 1.3.x (POWER 5.2)	ESA 2.1 (POWER 6.1)
Direction(s)		
Forward	X	X
Backward	-	X
Starting points		
Top	X	X
Current line	X	X
Bot	-	X

For skips, POWER dynamically uses the best entry point out of begin-of-file/end-of-file/current-position

í more intelligent file access

- Applies to any scrolling action (e.g. IUI, PSF, DWF, CICS RCF). For SYSIN, SYSLST (]), SYSPUN, in Nonshared and Shared POWER

„ Benefits:

- The higher the current 'distance' to begin-of-file, the bigger is the I/O and the RT benefit vs VSE/ESA 1.3

No improvements vs VSE/ESA 1.3 shows up (same DBLK size) if viewing is/was without skips

„ Performance Observations

- Much faster scrolling for those cases where expected:

e.g. Scroll backward from the end of a 20K lines job

Elapsed time: 3 times less (if DASDs uncached)
25% less (if 100% cache READ hits)
I/Os: 7 times less
CPU-time: 3 times less

POWER Queue File Improvements ...

Ù Parallel Browse of Queue-Entries

- When need exists to view POWER output entries concurrently

Benefits:

- Missing function, avoids waits

  Up to 255 concurrent users (nonshared POWER)
  Up to 15 concurrent users (per shared POWER)

Ù Improved Queue Entry Locking for non-shared POWER

Queue file disposition change not required on disk

Benefits:

- Reduces I/Os to POWER Queue File by up to 30%

New service for Queue File access

Ù Direct Queue Entry Access

Provision of Direct Queue Entry Number in Spool Access Control service

Allows faster queue entry access (less CPU-cycles to search for entry within RDR, LST, PUN or XMT (direct only) queue) It also resolves a 'uniqueness' functional problem.

Refer to 'VSE/POWER Application Programming', SC33-6636

POWER Printing Improvements

POWER Output (2.1.0)

Ù **Increased Output Buffer Size**

4K output buffer to real U/R devices (printer), instead of 2K

Benefits:

- Reduces I/Os to physical POWER printers by up to 50%

Performance Observation for Printing:

- 9221-170 with cached 9345 DASDs
- VSE/ESA 2.1.0 vs VSE/ESA 1.3.6
- DBLK=4080 in both cases (default)

Activity	Workload	Elapsed time	Total CPU-time	Total I/Os	
Printing	-	+-0%	* -10%	-30%	
* Print time determined by printer					

POWER Printing Tuning Possibility

„ Background

Channel 9/12 controls in FCBs

Used to control maximum text and page size
(end of user and end of physical page),
required only if no line counting done by application.

VSE system components do line counting by their own,
elder applications may not

„ Tuning Action

Change programs to use private line counting

instead of PRTOV or 'at-end-of-page'

Remove Channel 9/12 from FCB(s) if really not required

Avoid unnecessary S/W interrupts and pathlengths:

- Unit Exception and Unit Check
- Sense
- Re-issuing of SSCH to printer

Proceed with care, avoid that any application run later on
causes skips to channel 9/12 in the spool output.

„ VSE/ESA 2.2 Enhancements (VSE/POWER 6.1.2)

Apply APARs DY44258 (UD50168, UD50170), replaces DY44177
DY44311 (UD50183, UD50187, UD50188)

to reduce S/W impact of such unnecessary interrupts during
actual printing (on any type of printer).

If required for any reason, channel posting can be
re-enforced by the VSE/POWER autostart statement 'SET
CH9CH12=POST'

POWER Printing Start Option

Ù PSTART LST,00E,class,buf

class is the POWER LST class to be started

buf specifies the number of input and output buffers used for printing:

buf	#input buffers (size = DBLK)	#output buffers (size = 4K)
1	1	1
2	1	2
D	2	2

Input buffers

- contain 1 DBLK each
- are filled through READs from POWER Data File (even ahead if 2 available)
- reside in partition GETVIS

Output buffers

- have been increased from 2K to 4K
- are written with EXCP REAL to the printer (each buffer with its own CCB)
- reside in POWER program area (PFIXed)

buf-values greater than the default 1 (i.e. 2 or D) make POWER printing more independent from any activity in VSE with currently higher priority than the POWER LST task.

The introduction of huge DBLK sizes somehow has reduced the need for 2 input buffers.
2 output buffers may be beneficial for very fast printers.

Performance:

No impact expected on CPU-time required for printing.

Print speed is less degraded by some higher priority work.
No improvement can be achieved by more buffers, if a printer is running at 'rated' (=full) speed.

POWER Printing Improvements ...

POWER Printing Enhancement (PTF)

Ù Offload LTA if Printer Not Ready (VSE/ESA 2.2 time)

„ History

If a physical printer was not ready for loading a new FCB, the LTA was blocked by the transient $$BATTF2 during '0P08A Intervention Required'.

„ Enhancement

Before the LTA is occupied, it is checked separately whether the printer is ready.

This avoids blocking the LTA, which is a VSE wide required resource.

APAR DY44459 with PTF UD50253 and UD50259

Unfortunately, the chosen method does not bring benefits for printers which can process NOPs even when not ready:

- 3211
- NIPSON 6110, emulating 3203-5
- ...

POWER Statistics

As is valid for all POWER releases:

Ù Get and check POWER statistics

- at POWER shutdown
- via D STATUS

and do a virtual storage tuning

Refer to separate POWER charts.

Bigger DBLK defaults (VSE/ESA 2.3.0)

VSE/POWER 6.3.0

Ù Bigger DBLK default sizes (4080 byte -> 7.5K)

The old DBLK default for all still supported DASD types was 4080.

In order to
- reduce the number of POWER I/Os
- even better exploit track capacities,

the following new defaults have been introduced and are used, except the user overrules the DBLK=O specification in a private POWER generation.

DASD type	3375	3380	3390	9345	FBA	
DBLK default	6816	7476	7548	7548	7680	

To avoid higher requirements for DASD space for the POWER Data File, the DBLKGP default value was reduced from 10 to 8.

Performance Impact

In general the number of POWER spooling I/Os (default situation) can be reduced by a factor of

up to new_default / 4080

The overall reduction of #I/Os depends on the relative spool intensiveness of a customers workload.

Example:

Let's assume,
- 10% of I/Os so far were to the POWER data file on 3390 DASD
- the DBLK size used changed from 4080 to 7548.

-> POWER Data File I/Os ared reduced by a factor of
up to 7548/4080 = 1.85

-> The relative number of total I/Os reduces up to
0.9 + 0.1/1.85 = 0.95

i.e. up to about 5% of total I/Os can be saved

POFFLOAD Pickup Function (VSE/ESA 2.3.0)

Ù Backup all entries w/o locking the entire system

This new backup function was created in order to more flexibly bring POWER queue members to tape, i.e. without locking the whole system for a certain time.

It is very similar to POFFLOAD BACKUP, but

- does not lock the whole POWER queue and thus the system
- is not a true point-in-time copy (snapshot)

The following table compares the different POFFLOAD options

	BACKUP	SAVE	PICKUP
Scope of lock	Entire system	Single spool entries	as SAVE
Type of OFFLOAD	True snapshot (system locked)	No snapshot (system runs)	as SAVE
Spool entry selection	All entries	Only dispatchable entries	as BACKUP (D,K). but *1 (NOT DISP=*,H,L, no time event scheduling job)
Disposition handling in POWER Queue	No change	Removes DISP=D, K to L changes	as BACKUP
Parallelism	Only 1 BACKUP cmd **	Concurrent SAVE cmds allowed **	as BACKUP **

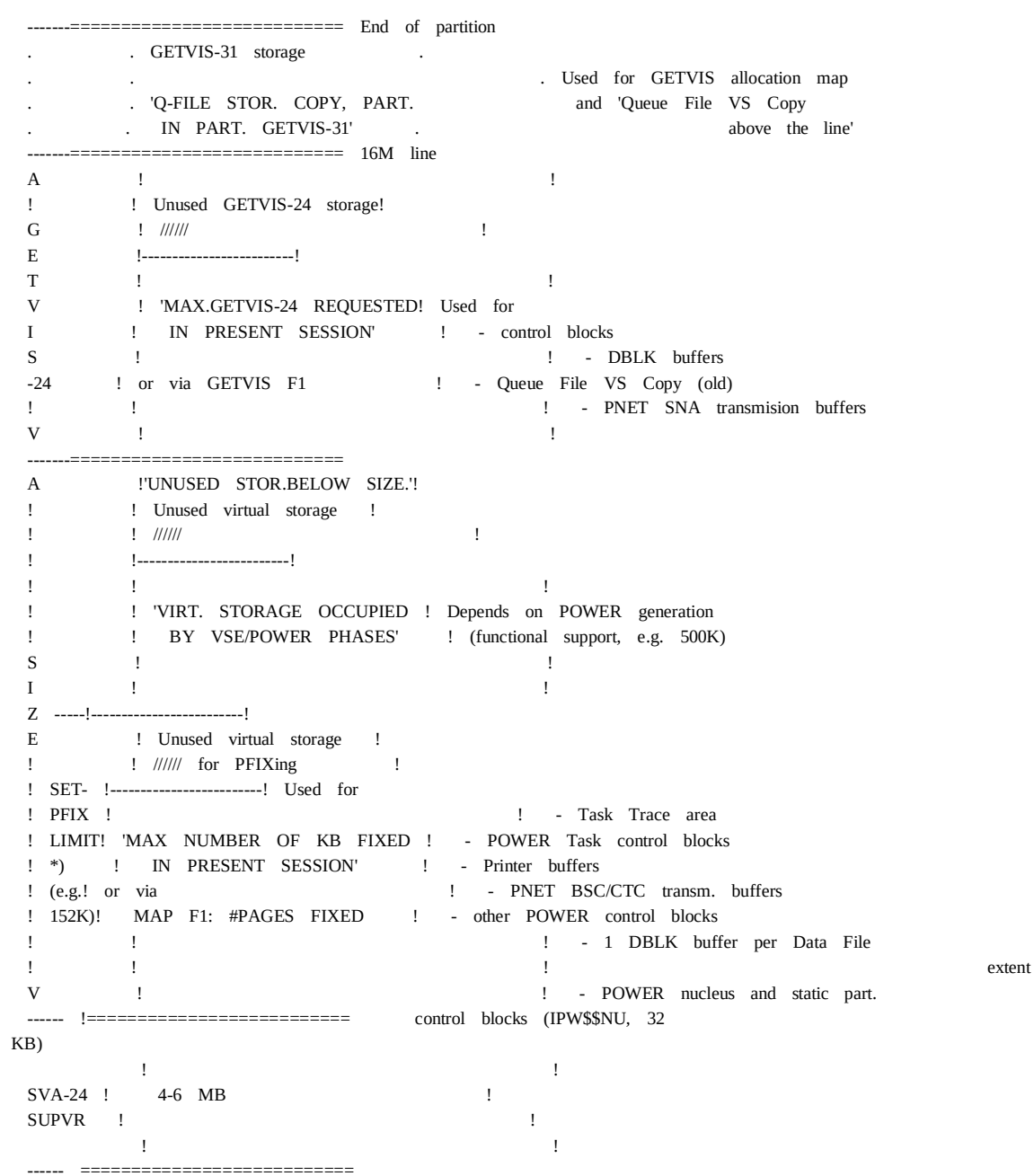
*1 PICKUP tries a real snapshot by flagging quickly all entries
- Entries deleted meanwhile (before 'real backup' is done) are/cannot be put on tape
- Entries added after PICKUP cmd time are NOT selected

- PICKUP is slower than BACKUP, since for each entry WRITE access to the POWER Queue File is required anew

** BACKUP and PICKUP cannot be used concurrently

POWER Virtual Storage Map

Values contained in output of 'PDISPLAY STATUS'



*) Maximum value for POWER PFIX allowance is 1M (2M in 6.4)
(In contrast to other use, it costs virtual storage if value is
higher than required)

POWER Virtual Storage Tuning

Refer to Virtual Storage Map on last foil

1. Collect the static data from your system

ALLOC = V-SIZE + GETVIS from MAP (or better MAP F1)
SETPFIX from MAP REAL
SIZE from // EXEC POWER
GETVIS from MAP

2. Collect the dynamic (=peak load dependent) data for a typical POWER session

3. Determine the unused portions

4. Check your actual GETVIS requirements

and roughly compare to the requirements shown here

5. Adjust the values determining POWER Virtual Storage

Unused areas (///// in previous foil) should not be too big,
if virtual storage below the line is a problem.

POWER VS Requirements

POWER PFIX Requirements

Note that

- in PNODE MAXBUF, 3 to 5 buffers each are enough
- each Data File Extent (<16) needs DBLK bytes

POWER GETVIS-24 Requirements

Major GETVIS-24 requirements are for

- **Spool/print/transfer data buffers**

  Refer to separate chart

- **POWER Queue File Virtual Storage Copy**

  Old, VSE/ESA 2.4 and POWER PTF allow residence in GETVIS-31

- **Misc. control information, etc.**

To fully exploit POWER Virtual Storage ...

- **Use the Queue File above the line**
- **Select a DBLK size of say track/2**

POWER GETVIS Requirements

Ù Spool/print/transfer data buffers

As a rough estimate, the space requirements are a direct function of the DBLK size
(neglecting any buffer control info, any rounding):

#tasks x DBLK

# Started POWER Tasks	
Local tasks	writer / punch card reader tape reader
SAS connection function	reader / writer
RJE tasks	reader / writer
Started partitions, for Execution ...	reader / writer
PNET job or output	transmitter/receiver

In practice, you get the actually used total value in

'MAX #TASKS ACTIVE AT ONE POINT IN TIME' from D STATUS

Rough estimate:

1 spooled partition in average uses 2.5 POWER tasks

EXAMPLE: 100 started POWER tasks need about	
DBLK= 4080	0.4M
DBLK=14000	1.4M
DBLK=28000	2.8M

More Info

Refer to

'VSE/POWER Administration and Operation',
under 'Fixable and Virtual Storage Requirements'
- V6.1 p29, SC33-6633-01
- V6.4 SC33-6733-00, 06/99

POWER GETVIS Requirements (cont'd)

Ù Queue File Virtual Storage Copy

	VSE/ESA 2.3 (POWER 6.3)	VSE/ESA 2.4 (POWER 6.4)
Bytes req'd per POWER job (Queue Record length) 256 byte	368/384 byte	
Queue Record Block length 4096 byte	12288 byte	
#entries per Q.Rec. Block 16	32	
VSE/ESA 2.4 increased the Queue Record length to 368 byte, residing in a 384 byte compartment (= I/O blocksize)		

Approximate requirement:

```
#Queue_Records_In_Queue_File_On_Disk
------------------------------------ (KB)
                                4 or 2.67
```

Check for your environment, via DISPLAY STATUS

'TOTAL NUMBER OF QUEUE RECORDS'
and 'QUEUE FILE STORAGE COPY PARTITION GETVIS SPACE'

EXAMPLE:	1000 queue entries	0.25M or	0.375M
	5000	1.25M	1.875M
	10000	2.5 M	3.75 M
	32766 (max)	8.0 M	12.0 M

By the way, when analyzing customer dumps, we saw up to 20000 POWER queue entries]

NOTE: There is no need to increase the POWER Queue File on DASD when the POWER Data File is increased. (A Queue File entry is related to a POWER job, independent of its size, i.e. the number of DBLKGPs required).

'POWER Queue File above the line' was also retrofitted to VSE/ESA 2.x (APAR DY44864, PTF UD50664/UD50665, 08/98). In addition, the new autostart statement

SET GETVQFL=ANY

is required, for VSE/ESA 2.x, NOT for VSE/ESA 2.4.

POWER Queue File in VIO

For customers with a huge POWER Queue File requirement, which are not (or not yet) being able to put the Queue File virtual storage copy into GETVIS-31, an alternative with VIO exists.

- **Some customers tend to have a HUGE POWER Queue File**
  - **Usual system growth**
    More partitions, more throughput
  - **Keeping LST output e.g. for 1 week and more**
    E.g. to be prepared for re-claims of output jobs
  - **POWER Queue File does not fit in remaining GETVIS-24**
- **APAR DY44496 (PTF UD50325, UD50326) allows to use VIO instead, as a circumvention:**

  Retrofitted to VSE/ESA V1.4 as DY44428, standard in VSE/ESA 2.3.

  You can enforce to place the Queue File virtual storage copy into VIO by the AUTOSTART statement

  SET QFILE=VIO

  (POWER in shared or in private space).

  Make sure that enough VIO is specified in the SUPVR statement of the IPL procedure. If you specify much more than needed, it just costs DASD space for the page data set, nothing else.

- **Queue File in VIO no more required if in GETVIS-31**

  (Except you would run (RRRGGGG]]]]]) VSE/POWER in a shared partition)

POWER Queue File in VIO ...

Ù POWER Queue File 'location'

If SET QFILE=VIO is not set, POWER will place the POWER Queue file as follows:

POWER in ...	Sequence of trials
Private space	1. In Partition GETVIS (if sufficient) 2. In VIO
Shared space *1	1. In VIO (if sufficient) 2. In Partition Getvis
*1 POWER in shared space would be required for old PUTSPOOL/GETSPOOL usage - With VSE/ESA 2.4 (or VSE/ESA 2.x + PTF) it is always possible to exploit sufficient GETVIS-31 space	

With the POWER Queue File VS copy above the line, D STATUS also gives

QUEUE FILE STOR.COPY PART IN PART. GETVIS-31 nnn K-BYTES

telling how much of it is actually occupying GETVIS-31 space.

Vendor products that do NOT access queue records by standard VSE/POWER interfaces (]), must be prepared for AMODE-31 addressing.

For debugging purposes the following autostart statement is possible to limit the queue file to GETVIS-24

SET GETVQFL=24

even though enough GETVIS-31 would be available.

Ù CPU-time impact

CPU-time increase VIO vs GETVIS usage	
Average Queue File intensive	up to +2%, and more
Queue File intensive load	
VPOOL=64K	up to +6%, and more
VPOOL=256K	up to +2%, and more
- CPU-time increase is proportional to the number of VIO POINTs, which may increase a lot when 'the Queue File in VS is fragmented across pages'	

POWER DBLKGP Trace

Function:

„ Checks migration of DBLKs from used to free chain and vice versa of the POWER Data File.

Permanently verify the consistency of all DBLKGP chains.

Overhead:

„ Causes more I/Os (1 READ and 1 WRITE per DBLKGP) and some increase in CPU-time when DBLKGPs are freed (i.e. upon DELETE of a POWER Queue entry).

Relative overhead is smaller for bigger DBLK group sizes (= DBLK x DBLKGP, the smallest unit for DASD space allocation).

Actual measurements showed an overall DBLKGP trace overhead of 1% CPU-time.
When deleting power queue entries, the overhead in I/Os (see above) may reach a higher but very temporary impact.

Control:

„ Status of the DBLKGP trace is displayed via D STATUS

DATA FILE DBLK GROUP TRACING ENABLED or DISABLED

By default this trace is on (enabled).

í **Once your system is in stable state, you may set it off via**

PSTOP DBLKTR

- as a POWER command (valid until next POWER cold start).
  This is recommended, but do not forget to reenter the command.

- in the POWER startup.
  Note that a change back to DBLKGP tracing in this case is only possible via a new POWER coldstart w/o this line]

POWER Task Dispatching Trace

Function:

„ Trace task history for trouble shooting purposes.

Every task dispatching event is recorded in wraparound fashion in the trace area.

Overhead:

„ <1% CPU-time overhead for avg spool intensive batch loads

Control:

„ Status of the Task Dispatching trace is obtained via

D TRINFO

„ By default is active, since by intent started via PSTART

í **Leave it on if any problem should occur (our recommendation),**

but you could set it off via

PSTOP TASKTR

POWER Accounting

„ Function

Collect POWER-job relevant data for Accounting purposes:

- execution account record (once per job step)
- list account record
- network - " -
- reader - " -
- punch - " -
- ...

The counting itself is done in any case. For PACCOUNT=YES, each account record is written in 1 SSCH to the POWER ACCOUNT FILE. Overall, this may end up with about 3 SSCHs per POWER job

„ More Aspects

POWER Accounting is based on VSE Job Accounting, and thus needs JA=YES.

The product default is PACCOUNT=NO, but PACCOUNT=YES is the shipped default.

„ Measured POWER ACCOUNT Overhead

Using the average spool and job-step intensive PACEX VSE batch workload,

about 1% CPU-time

was saved when switching POWER Accounting off.

„ Recommendation

Set PACCOUNT=NO for the (seldom) case that you should NOT use the POWER Accounting function

Refer also to the POWER tuning hints in the 'VSE/ESA 1.1/1.2 Performance Considerations' document

POWER and VSE Partition Priorities

„ Some Technical Background

Ù VSE/POWER's major activities

- provide spooling services for any VSE partition
- control dynamic partitions (i.e. partition classes)
- provide additional services
  (e.g. local printing, PNET, ...)

Ù General rule

It is always the standard (and most often a vital requirement for multi-programmed systems) to give such a 'central' partition higher processing priority than any of the controlled partitions.

Ù POWER priority setting

Customers occasionally wish and do put certain POWER controlled partitions 'above' POWER, e.g.

- VTAM
- TCP/IP
- Performance monitors
- CICS.

This change in PRTY setting may to some (even smaller) extent reduce the impact of 'high' POWER CPU consumption (e.g. high spooling or printing) on a 1-way e.g. on CICS response times.

POWER and VSE Partition Priorities (cont'd)

„ POWER NPC Parameter

Ù Setting

To allow such a PRTY setting (and still to have the priority check done by VSE), the NPC (No Priority Check) parameter was introduced for STATIC partitions:

PSTART part-id,classes,NPC

Without NPC, no static partition with higher or even same priority can be PSTARTed (includes even same priority partition if partition balanced). In such a case, msg 1R63I is being issued.

Ù Dynamic partitions

For DYNAMIC partitions/classes NPC is not available/required, since

- they are 'PSTARTed'/controlled by POWER anyhow (depending on the Dynamic Class Table).
  No explicit PSTART is required, 'NPC is used internally'

Ù PRTYIO

Note that I/O priorities are not affected by partition priority or NPC setting.
PRTYIO is described in the I/O Performance document.

í In case of disk volume contention, use PRTYIO first

POWER files on same volume(s) as other Online work (and higher logical device contention)

í You may specify NPC for exceptional partitions, and 'activate' it via PRTY only when required

It may help to a small extent in a specific situation

POWER Shared Spooling

Ù Share POWER Queue and Data spool files between up to 9 VSEs via DASD Sharing

Load Sharing/Balancing:

A POWER job can run on any participating VSE system, provided the POWER class is started/available (and SYSID specification allows)

Flexible Use of Resources:

Execution and Printing of a job can be done on separate VSEs (e.g. a fast printer only available on 1 VSE system)

Jobs can be executed without data movement on that VSE system which has access to required data.

Ù Implementation

„ Concurrent update of the POWER files is avoided by using exclusive WRITE access.
Can only be held for a certain maximum time interval (t1)

„ Other VSE systems requiring exclusive WRITE access are queued (Naturally, any function not requiring exclusive WRITE access can continue)

„ WRITE activities against the POWER Data File (e.g. spooling job input or output) can always continue until the current DBLK group is filled

„ Read activities (e.g. printing of a job) can continue until EOJ

„ At the end of each exclusive WRITE interval ...

- Collected change info is written to the Queue File on DASD

„ At the begin of each exclusive WRITE interval ...

- Change info is read from the Queue File on DASD in an efficient way (queue record blocks)

POWER Shared Spooling Hints

- **Leave SHARED=NO (default) in the POWER macro, if POWER files are not shared**

- **All POWER Queue file hints apply as for non-shared POWER**

  Of specific value are bigger DBLK groups (DBLKxDBLKGP)

- **Be aware of the TIME=(t1,t2,t3) values in the POWER macro.**

  Defaults are (5,0,60) and usually adequate.

  t1 Active Time: Maximum interval during which 1 POWER can keep exclusive update access. Control is given back as soon as no more required. (1 to 99 sec)

  t2 Idle Time: Interval POWER has to wait until it is allowed to issue another update request. (0 to 9 sec) Can assure that a slower processor gets enough share.

  t3 Polling Time: Time interval after an idle VSE/POWER re-checks the POWER queue for new work (1 to 999 sec)

  Values need not be identical for all participating POWERs

- **Move POWER files away from system volumes, if they are busy with non-POWER activities**

  Refer to 'Recommendations for DASD Sharing'

POWER PNET Considerations

Ù Function

Connect VSE/POWER with other VSE/POWER or other platforms supporting the NJE (network job entry) protocol.

Transmit jobs and output data (besides commands and messages) between job entry nodes, connected in a network, via BSC lines, CTCAs, SDLC lines (SNA)

PNET Performance Dependencies

„ Size and number of PNET Transmission buffers

PNODE generation macro: BUFSIZE=nnnnn, MAXBUF(n-rcv,m-tsm)

Buffers reside in PFIXed part of the POWER partition (BSC and CTCA), in GETVIS-24 for SDLC.

Í Specify reasonable BUFSIZE on BOTH sides of a connection

For BSC lines, if unreliable, a high BUFSIZE may lead to more frequent retransmissions. For CTCA, use BUFSIZE as big as possible.

MAXBUF specifies the number of buffers per PNET receiver and transmitter task.

Default is (3,3), minimum is (2,1). The maximum is (255,255), but usually up to 8 buffers should be sufficient.

Í You may specify higher MAXBUF

- when line speed is higher
- when traffic volume is high

For SNA you may use

MAXBUF(xmit bufs in adjacent node, VPACING in VTAM B-book)

POWER PNET I/Os	
Accesses to POWER Data File	1 DBLK per disk-I/O
to BSC line/CTCA	1 BUFSIZE per I/O
to VTAM (PNET SNA)	1 BUFSIZE per SEND/REC.

POWER PNET Considerations ...

PNET Performance Dependencies (cont'd)

„ Number of concurrently active PNET transmitters and receivers

Starting more transmitters and receivers may not improve overall PNET throughput, if line/network capacity is reached already

„ VTAM SNA parameters (Pacing, Buffers)

- a low value of pacing (e.g. 3) saves virtual storage
- a higher value of pacing may enhance PNET speed/capacity if
  - sufficient VTAM LFBUF and VPBUF buffers
  - line capacity

  is available.

í Make sure that for PNET SNA enough DSPACE is defined in the VSE/POWER startup

POWER PNET SNA is a VTAM application.
One Dataspace is shared with VTAM for all VTAM applications within POWER.
Usually // EXEC IPWPOWER, DSPACE=2M should be sufficient.

„ Performance of POWER Queue/Data File access

Includes DBLK (business as usual, refer to other charts)

„ Status of PNET Traces

- BSC/CTCA Console Trace
- RJE,BSC and PNET Telecommunication Trace
- VTAM buffer trace

Required only for functional trouble shoooting

„ It is not expected that PNET performance is such that it could effectively replace POWER Shared Spool

POWER PNET Enhancements

Ù Increased Max. PNET Transmission Buffer Size (VSE/ESA 2.1.0)

BUFSIZE value in the POWER PNODE macro, default is 400

4000 vs 1800 bytes for PNET BSC line or CTCA support

Benefits:

- Reduces I/Os to transfer data by up to 50% (1 I/O per buffer)

Maximum BUFSIZE for PNET SNA is 32K

Ù Bigger PNET Buffer Size Default (VSE/ESA 2.3.0)

The increase of this value from 400 to 1112 bytes

- reduces PNET I/Os
- and thus the number of VTAM Send/Receives or CTC/BSC line I/Os

It becomes effective when PNET is activated with a newly generated Network Definition Table (NDT), in which the PNODE macro does NOT specify BUFSIZE=.

In any case, try to specify a bigger BUFSIZE value.

POWER PNET vs Shared Spooling

Ù Function

„ PNET can span systems outside the scope of DASD Sharing

„ Shared Spooling is more flexible and provides a single view

In those cases, where both would be applicable functionwise
(i.e. where the additional management of jobs in PNET can be done),
performance becomes relevant.

Ù Shared Spooling Performance

- Minor CPU-time increase and some I/Os on top
  compared to Non-shared POWER

- Capacity of a VSE/POWER may be sooner exhausted, since more
  jobs in average
  (relieved by POWER Queue File above the line).
  Manual control of POWER Queue gets more complex

Ù PNET Performance

- Some CPU-time increase and quite some I/Os
  to move the jobs and data physically between DASDs

- Data movement is only acceptable, if VCTC can be used
  (i.e. under same VM), at best with PNET CTC,
  maybe without VTAM

Ù Alternative for mult. VSEs under VM or in LPARs:

Use a combination of both

E.g. in a complex with 8 POWER systems, use

2 4-fold Shared Spool systems connected via PNET
instead of 1 8-fold Shared Spool system
or 8 non-shared POWERs connected via PNET

Shared spool complexes must be able to share DASDs

PART F.

CICS/VSE 2.3

CICS/VSE 2.3 Performance Overview

- CICS functions equivalent to CICS/MVS 2.1.2+
- For all CICS 2.3 related performance information, refer to

  'CICS/VSE 2.3 Release Guide and Performance Guide'
- No increase in static virtual storage

	CICS 2.2	CICS 2.3
CICS nucleus modules 840K CICS resident modules 132K	824K 134K	
Total	972K	956K
- Results from HURSLEY, CICS DSW workload		

No CICS code moved above the line (yet)

- Support of LE/VSE: COBOL for VSE, PL/I for VSE, C for VSE
- Minimal CPU-time increase by LOAD of CICS programs

in order to check LE/VSE language usage
- loading resident CICS programs at CICS startup
- loading non-resident programs, when CICS is up

CICS/VSE 2.3 Specific Performance Results

VSE/ESA CICS/VSE	1.3 2.2	1.4 2.3	2.1 2.3 (*)	
Tot.CPU-time Base		+1%	+4%	

CICS/VSE 2.3 results on top of VSE/ESA 1.3

- **Various transaction workloads were measured**

  (Assembler, COBOL/VS, COBOL II, PL/I)

  Delta in CPU-time was below 1% to 2% in all cases

  (CICS/VSE 2.3 vs CICS/VSE 2.2, both on VSE/ESA 1.3)

  Runs also included MRO transaction routing, and CICS startup and restart times.

- **This delta (*) applies for VSE/ESA 1.4 vs 1.3 CICS workloads**

CICS/VSE 2.3 within VSE/ESA 2.1

- **Measurement results (RAMP-C)**

  Delta in CPU-time was about 4% overall per tx

More Info

For CICS/VSE specific component performance refer to

- GG24-3857, CICS/VSE Performance and Tuning
- SC33-0703-02, CICS for VSE/ESA: Performance Guide, Release 2.3, April 95, 332 pages
- **CICS Shutdown Checklist**

  Contained in 'IBM VSE/ESA Hints for Performance Activities'

Some CICS 2.3 Performance Enhancements

„ New SIT parameter DTB=AUX,FORCE

CICS 2.2 DTB GETVIS-31 problem solved:

Potentially 'huge' peak DTB space requirements for SAP R/2.
Amount was very hard to determine since very data dependent.
Problem occured even with DTB=AUX and GETVIS-31 (*)

If dynamic log buffer full (DBUFSZ), DTB records are spilled to ...

		DTB records spilled to
DTB=MAIN	(default)	24-bit space
DTB=AUX	(2.2 w/o GETVIS-31)	TS-AUX
	(GETVIS-31 avail.)	31-bit GETVIS(*)
DTB=AUX,FORCE	(CICS 2.3 only)	TS AUX (**)

(**) Select a big CISIZE for TS

„ CICS EXEC call allows change of tx-priority

CEMT INQ TRANS PRIORITY still available
to change transaction priority by operator (overtype)

EXEC CICS CHANGE TASK PRIORITY xxx
can be issued by a program to dynamically change its priority
(transaction source code must be available)

Allows CICS transactions a situation dependent 'social behavior':
transactions casually being 'long running transactions'

CICS 2.3 Performance Enhancements (cont'd)

„ New SIT parameter ISC=(YES,NOFS)

In CICS 2.2 and 2.3 some CPU-time overhead is possible in a TOR/AOR by ISC=YES (caused by unnecessary table lookups). An ISC=YES specification is required for interregion (MRO) and intersystem (ISC) communication:

	ISC=YES req'd in all CICSs	,NOFS applicable *)
Transact. Routing (TR) Y Function Shipping (FS) Y (VSAM, DL/I, TS, TD)	Y N	
Distr.Txn Process.(DTP) Y Distr.Pgm Link (DPL) Y (EXEC CICS LINK,remote) Asynchr. Processing (AP) Y (EXEC CICS START,remote)	Y N N	
*) ,NOFS is only possible for pure TR or DTP setups. Not possible if participating in FS, AP, or DPL, i.e. where 'transformer 1-4' code is required		

The VSE/ESA 2.1 workdesk ('VSE/DWF') needs ISC=YES for LU6.2 DTP.

Refer to APAR PN70494, the CICS PTFs are UN76443 and UN76442

í Reduced overhead by ISC=(YES,NOFS)

Sample measurement results

- Extreme case of a CICS transaction, doing EXEC CICS READ FILE (FILEA) only (in a loop)

Relative CPU-times for test-tx				
	ISC=YES	ISC=(YES,NOFS)	Delta	
FILEA as VSAM KSDS 208	sec	149 sec	- 28%	
FILEA as CICSTABLE 108	sec	26 sec	- 76%	
- Results from H.J. Ebert, FSC Munich - Figures shown apply to TRACE=ON, TRACE=OFF deltas are even higher				

CICS Internal Trace - Tuning Potential

CICS Internal Trace

„ Purpose

Required for test-CICS systems

- for debugging purposes

Recommended for production

- for the starting period only,
  until all new components/products/applications tested

May be used permanently

- to improve trouble shooting chances
  for very situation dependent functional problems

„ Control

TRACE=(nnn,ON) in CICS/VSE SIT table.
Default is ON

CEMT transaction also can control Internal Trace:
via CEMT SET TRACE ON!OFF

„ Performance Impact of TRACE ON

Workload	CPU-time cost	NPS change	
DSW workload (CICS funct. intensive)	+ 18%	0.25 -> 0.21	
RAMP-C workload (VSAM file intensive)	+ 8%	0.29 -> 0.25	

- Response time impact is very dependent on CPU utilization

- Results for VSE/ESA 2.1, also valid for VSE/ESA 1.3/1.4
- Non Parallel Share (NPS) of interest for Turbo Dispatcher
- When CICS monitors are active, figures may differ

í Switching CICS Internal Trace off may save a lot of CPU-time

CICS Internal Trace (cont'd)

„ CICS Internal Trace and CICS monitors

Be aware that CICS monitors may 'activate' the CICS Internal Trace, even if formally switched off.

This is done in order to utilize the CICS Trace hooks for monitoring purposes (i.e. to exit to the monitoring program). Only when the CICS Trace is really set ON by the user, control is on top given to DFHTRP, as is done for the CICS Trace w/o CICS monitors.

Í Be not astonished if you have set CICS Trace OFF, but CICS CEMT INQ TRACE tells you that CICS Trace is ON

EXPLORE/CICS users:

Make sure you have the newest performance PTF for EXPLORE/CICS 6.51 installed:

ES65106

Contact Computer Associates for more information.

„ CICS Auxiliary Trace

Refer to 'VSE/ESA Hints for Performance Activities'

CICS MRO Enhancement

„ Improved Performance for CICS/VSE MRO Fct Shipping

This APAR improves CICS journaling and reduces TD Non-Parallel Share in case of CICS Function Shipping:

APAR PQ13099 PTF UQ19908

The PTF causes changes in DFHIRP (reduced key 0 usage) and in DFHSPP and DFHSPZ (reducing number of logging requests, as reported in CICS/MVS APAR PN02707)

PART G.

ACF/VTAM 4.2

VTAM 4.2 Performance Aspects

Summary

- **31-bit Support for VSCR**
  - **VTAM address space**
  - **VTAM data spaces**
    - **with Cell Pool Management**
      to manage data space storage
      - **For VSCR**
      - **For new functions**
        APPN networking
        Persistent sessions (not supported by CICS/VSE)
        Data compression
- **Hardware Data Compression**
- **VTAM Internal Trace Enhancement**

  VTAM Internal Trace now active all the time
  (basic 5 options: API, SSCP, PIU, NRM, MSG)

  - DUMPs always show basic VTAM activity
  - Enhances serviceability ...
    at some cost for VTAM pathlengths

VTAM Trace option	Location of trace data (basic 5 options)
VTAM Trace OFF or MODE=INT	SVA-31 (wrap around)
VTAM trace MODE=EXT	SVA-31 (temp. trace area) VTAM Trace File
- 50 pages in SVA-31 (100 2K buffers vs 32 before) are now the minimum for tracing	

- **APPC Full Duplex**
  - **Improved session performance**

VTAM 4.2 with 31-bit Support for VSCR

Charts constructed with Doug Trottman and Carol Ames, IBM RALEIGH

Ù VTAM 4.2 Virtual Space -Summary-

- „ **Only VTAM start code must reside below the line**

  Almost all VTAM 4.2 modules are RMODE 31 capable

- „ **All VTAM I/O areas and I/O control blocks reside below the line**

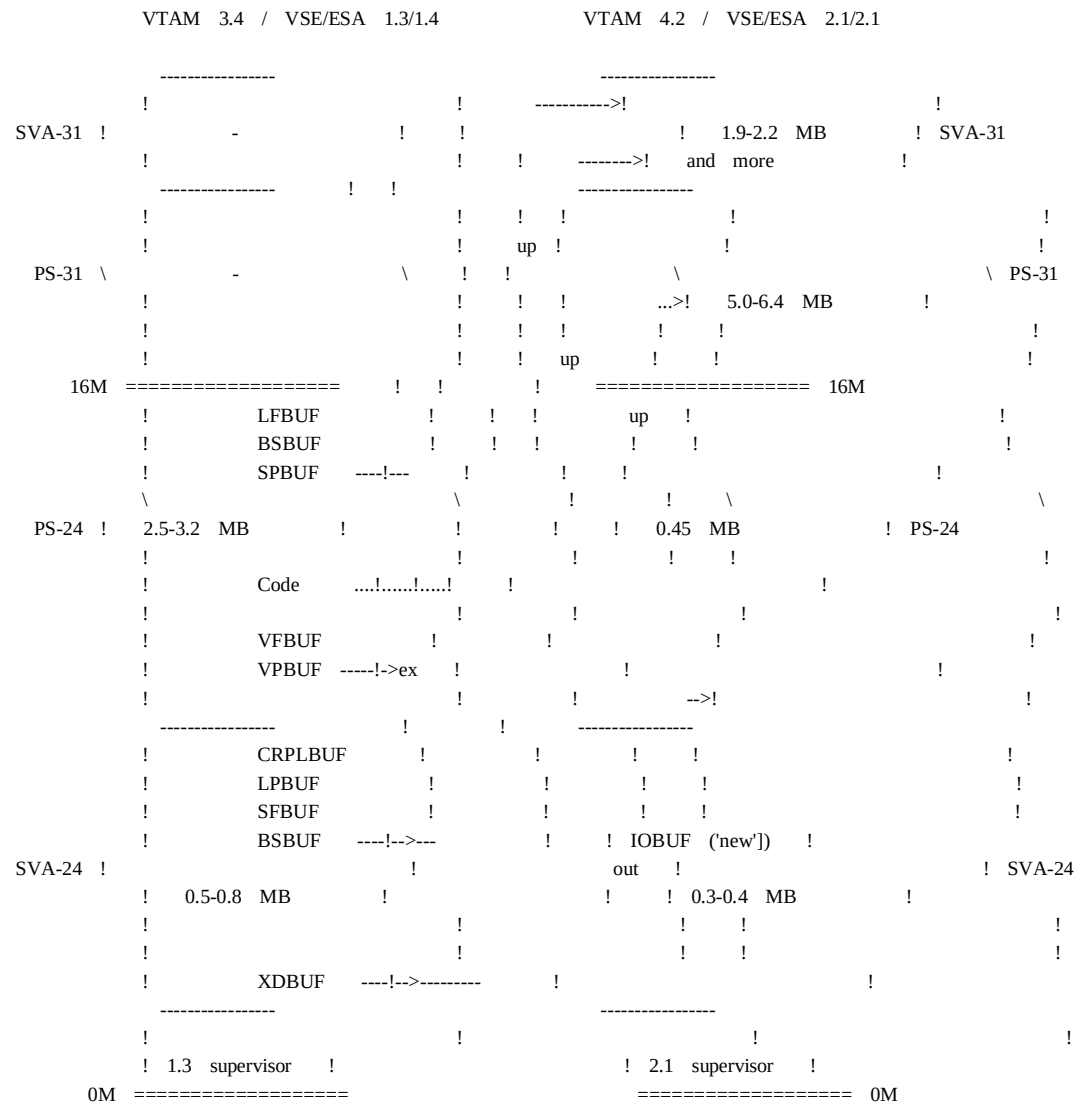
  A VSE restriction for CCWs and CCBs, the I/O data itself could be above the line.

  IOBUF buffers (I/O data and CCWs), IOBLOCK pool (CCBs)

- „ **All other VTAM buffers moved above the line**

- í **Higher VTAM capacity (more defined users, more concurrent sessions)**

- í **Less consumption of system-wide shared space-24 (VSCR)**



í **'Move it out' or 'Move it up'**

APAR DY43770 (VTAM 4.2 in VSE/ESA 1.3.0)

This APAR was required to avoid that initial buffers were created in SVA-24 instead of SVA-31. This happened

- if VTAM partition did not cross the 16M line

or

- if not enough SVA-31 GETVIS was available, but SVA-24 instead

VTAM 4.2 Storage Requirements

Ù VTAM 4.2 Virtual Storage Requirements

	Private below 16M	Shared (SVA-24)	Private above 16M	Shared (SVA-31)	
Modules	300K	32K	4000-5000K	800K	
Buffers *1	XDBUF +Pools 50K	IOBUF +Pools(incl. IOBLOCK) *3 100-200K	Pools 200K	CRPLBUF LFBUF LPBUF SFBUF SPBUF BSBUF +Pools 200-300K	
Other areas *2	100K	150K	800-1200K	1000-1200K	
4.2 Totals	450K	300 - 400K	5000-6400K	1900-2200K	
	about 7 to 10 MB, + data space(s)				
(3.4 Totals)	(2500-3200K)	(500 - 800K)	-	-	
	(about 3 to 4 MB, all space below the line)				

- All values vary with functions used and configuration and/or transaction rate

*1 Includes storage pools seen by using D NET,STORUSE
*2 Work areas for modules, permanent control blocks (i.e. all other non-module GETVIS storage)
*3 IOBLOCK = 1024 byte x #physical lines

- For VTAM 4.2 data space sizes, refer to separate table

í VTAM 4.2 shared-24 relief:

200-400K, includes 135K by VTAM code

VTAM 4.2 Storage Requirements ...

Ù Individual VTAM 4.2 Buffer Locations

VTAM Buffer Type	VTAM 3.4 (all below)	VTAM 4.2	VSCR contribut.	
CRPLBUF	SVA-24 p	SVA-31 p @1	++	
LPBUF	" p	" p @2		
SFBUF	" f	" f @3		
XDBUF	SVA-24 f	PS-24 f @4	++	
LFBUF	PS-24 f	SVA-31 f @5	+	
IOBUF	-	SVA-24 f @6	--	
BSBUF	PS-24 f	SVA-31 f @7	+	
SPBUF	" p	" p @8		
VFBUF ! expansion	PS-24 f -	@9	+	
VPBUF ! buffers	' p -	@9	+	
IOBLOCK pool	SVA-24 f	SVA-24 f		

PS Private Space f/p fixed/ pageable

@1 1 buffer for each concurrent VTAM appl. pgm request
@2 1 buffer for each active VTAM process
@3 1 buffer for each active application pgm
@4 XDBUF = 673 byte x #terminals (PU 2.1 ...), expandable
@5 LFBUF I/O buffers now are contained in IOBUF,
so LFBUF smaller in VTAM 4.2.
@6 IOBUF contains I/O buffers, expandable.
(were in LFBUF and VFBUF in VTAM 3.4).
Required number depends on PIUs/sec.
IOBUF must reside below the line in VSE.
Shipped values are (70,288,,,11)
@7 1 buffer for each concurrent boundary LU-session
@8 1 buffer for each concurrent LMPEO Send request
@9 Buffer type vanished. Some areas in data space,
most areas in VTAM partition GETVIS.
Expansions no more limited by VF/VP limits,
since handled via GETVIS/FREEVIS

Ù For more info refer e.g. to

'VTAM Networking Implementation Guide,
V4.2 for MVS, VM and VSE', SC31-6494 (page 583 ff)

Storage information is obtained by ISTSTATS VTAM application.
Data can be downloaded to OS/2, to be used as input to the
'VTAM Storage Estimate Program' SK2T-2007-001.

VTAM 4.2 Storage Requirements ...

ù Remaining VTAM 4.2 SVA-24 phases

Phase name	Size
ISTAICIR	1.6K
ISTAPCIN	2.3K
ISTAPCKU	1.6K
ISTAPCRS	1.5K
ISTAPCUE	5.2K
ISTINCFO	12.6K
ISTOCCCA	1.7K
ISTOCCOA	4.9K
Total	31.4K

Make sure that all eligible phases and buffers reside in SVA-31

... and not accidently in SVA-24
- if w/o VTAM APAR DY43770 and VTAM partition not above 16M
- if SVA GETVIS-31 and PSIZE-31 specification too small

ù VTAM partition size

Release	VTAM partition size
VTAM 4.2	about 5.5 to 7.0 MB + some increase of pools in private space may be helpful

Generous VTAM partition size may help,

though a much bigger size than used so far does not help.

Moving VTAM code in private space above the line brings only benefit if the private space below can be used for other VTAM purposes (i.e. is required).

Use a VTAM Partition size of say 15M

This would only waste some VSIZE on the page data set, but has the advantage that some GETVIS-31 space is available for few system functions.
(VTAM modules with LOC=ANY are loaded starting 'from above')

Use SIZE=ISTINCVT for VTAM 4.2

VTAM 4.2 Data Spaces

VSE/ESA Data Spaces Required for VTAM

	Min/init. size	#Data spaces	Expansion increment	Used for	
Base DS	1M	1	1M	VTAMs own APPN Control Point 6.2 application	
Add-on DSs $	1M	1/partition w/ VTAM APPLs	1M	Data queuing & Pers.sessions *	
Compression DSs $$ (optional)	2M	n, as req'd	not expandable	Compression	

- All these data spaces are owned/defined/allocated by VTAM

- Data space storage managed via 'cell pool management' (no overhead vs VSE GETVIS expected). Cell size varies from 120 byte to 64K

- Data queuing done for data w/o outstanding RECEIVE

* Persistent sessions (e.g. CICS restart) not supported by CICS/VSE 2.x

$ Data space created when first VTAM ACB in a partition opened. 2MB should be allowed via DSPACE=nM

$$ 1st data space created when a 'compression session' starts

- **Use QUERY DSPACE to get an overview on data spaces**

  Including total data space sums

- **Use QUERY DSPACE,ALL to get details on each data space**

  Including actual and maximum sizes (really used space is not seen)

Refer to Data Space Definitions on next page

VTAM 4.2 Data Spaces ...

Ù SYSDEF DSPACE,DSIZE=nM,DFSIZE=mM

Start with a high DSIZE

... to ensure that VTAM DSPSERV will not fail. A value is required, big enough for the max. sum of ALL data spaces (including Shared Data Tables for CICS TS):

All VDISK and Label Area requirements
All other data space requirements (vendor programs)
All Maximum values (DSPACE) for VTAM appls
VTAM DSPACE

Reduce carefully later, if you want (not required).
The shipped value is 20M (Environment B)

DFSIZE (default size of a dataspace) of 3M

1M is the system default, not modified when shipped. Use DSPACE below to determine MAXIMUM size for VTAM applications

Use PARTMAX=nn, if the default of 16 data spaces for VTAM should not be enough (e.g. many VTAM applications)

Ù // EXEC VTAM-appl,DSPACE=nM

„ The EXEC statement of any VTAM application (e.g. CICS, POWER PNET) should contain ',DSPACE=nM' to specify the maximum size of the expandable data space owned by VTAM:

',DSPACE=3M' is recommended due to data space(s) for data queuing.

„ If DSPACE is not spec'd, the SYSDEF DSPACE,DFSIZE value holds.
If also DFSIZE is not spec'd, the system default of 960K applies, which is rounded up by VTAM to 1M

Í For safety, use DSPACE=3M for production-CICSs

Ù // EXEC ISTINCVT,SIZE=ISTINCVT,DSPACE=4M

„ DSPACE here specifies the MAXIMUM size of VTAMs own data space ('Base DS'). Shipped value is only 2M.

VTAM 4.2 Fixed Storage Requirements

Ù SETPFIX LIMIT=(xxxK,yyyK) setting

- Amount of permanently fixed private storage

  System GETVIS fixing via GETVIS PFIX=YES does not require SETPFIX allowance

- Page frames are only 'stolen' from page pool if actually fixed

- SETPFIX LIMIT=(xxxK,yyyK) is required to allow permanent fixing of pages in private space (partition) via PFIX macro (fixing below and above the 16M REAL(]) line)

  Required for fixing of pages in VTAM partition
  - Modules
  - XDBUF
  - Misc. storage/control blocks

- VTAM 4.2 can deal with 31-bit REAL storage.
  Thus, virtual-31-bit should be fixed in real-31-bit storage

í Specify SETPFIX LIMIT=(,yyyK) besides SETPFIX LIMIT=(xxxK)

to avoid that 24-bit real storage is used, when 31-bit real storage could have been used (needs >16 MB real).
This, in general, does not seem to be a performance problem

Start with say (,400K). Use MAP REAL, reduce if you want.

Make sure you have applied PTF UD49781 for APAR DY43755.
It assures that PFIX-31 is used, provided VTAM partition crosses the 16M virtual line

VTAM 4.2 Fixed Storage

VTAM 4.2 activity	PS-24	SVA-24	PS-31	SVA-31	Total		
Start w/o any activity	32K*	136K		64K	260K	468K	
300 active RAMP-C terminals	**			**			

* Value w/o PFIX-31 allowance is about 64K higher
- RAMP-C with about 4.5 tx/terminal and minute

** Values did not increase by VTAM activity

Further VTAM Performance Aspects

Ù VTAM 4.2 Real Storage Requirements

Increased vs VTAM 3.4, but not as much as total virtual size (including data spaces) look like.

Increase is higher if ...
- more buffers exploited
- data compression used

Ù VTAM SEND/RECEIVE pathlengths

No marked delta expected so far.

A significant amount of code was moved to SVA-31 to allow the VTAM code to run under the USER task (like MVS).

This will also result in VSE JA deltas for CPU-times

í Do NOT consider CICS JA data (or DSA values) w/o VTAM CPU-times

Ù VTAM Performance for network and session activation

Make sure to have VTAM PTF UD49442 applied.

This PTF implements GETVIS subpools in order to get fast and less fragmented GETVIS service

Ù Let SGALIMIT=0 (default) in VTAM startup book

A value of 0 in ATCSTR00.B means unlimited VTAM System GETVIS usage (24 and 31-bit combined). This is in general appropriate and should only be changed, if during startup shared space shortage would occur.

The same applies to the limit for 24-bit only: SGA24

General VTAM Tuning Info

Ù Data Compression

TBD

Ù Monitor VTAM buffer pools with D NET,BFRUSE

Avoid frequent buffer pool expansions.

Increase number of buffers to 'MAX USED' if expansions occur.

Reduce number of buffers if 'MAX USED' much lower than 'MAX AVAILABLE'.

Do NOT change any buffersize, except IOBUF.

Especially observe IOBUF (since in SVA-24).

- Use TNSTATs to calculate average PIU size for VTAM SNA PUs.
- Use HOTIOTRM to limit IOBUF use by a runaway terminal.
- Do not use an IOBUF start option size of <256 byte (Info APAR II02008), try 480 byte.

For more details on tuning VTAM buffer pools, refer to
- VTAM Networking Implementation Guide,
- or to Jon vonWolfersdorf's paper.

Ù Use D NET,STORUSE (available since VTAM 3.4)

in case of storage problems

Check for pools with excessive storage requirements

The Network Implementation Guide (Part 5) explains what function each pool supports. That function may be turned off to relieve the storage shortage

Ù Monitor available System GETVIS-24 space

More info on VTAM 4.2 Data Spaces

The following text partly complements the previous considerations.
It is added here for further clarification (taken from II08816).

1. General

VTAM requires 1 MB for initialization, in addition, for each partition running VTAM applications 1 MB.
This would be for example if POWER/PNET through a VTAM line is used:

1 MB for VTAM, 1 MB for POWER and 1 MB for CICS

That is, in an environment with POWER/PNET 3 MB are needed as minimum, without POWER/PNET, 2 MB is the minimum since CICS always needs VTAM.
The first data space of 1 MB is created by VTAM itself, the second data space is created when the ACB is opened to CICS.
If there are user applications of VTAM active, for each partition running such applications, 1 MB has to be added.

2. Maximum Value for VTAM Data Spaces

There is a maximum size for the VTAM data space which may be specified either through the DFSIZE operand of the SYSDEF statement or by the DSPACE parameter of the EXEC job control statement. VTAM expands data spaces in 1 MB increments up to the specified limit.
The DFSIZE is a global value for all VTAM applications, this value is only for VTAM applications, private usage of data spaces is not affected. The value specified in the EXEC JCL statement is only for the related application and it overwrites the DFSIZE specification.
If DFSIZE is smaller than 1 MB VTAM will take 1 MB.

The IBM provided settings are: VTAM, CICS and POWER have a DSPACE parameter of 2 MB. DFSIZE is not used.
The maximum of 2 MB will be sufficient for most cases, if many different request unit sizes are used, we recommend to change the maximum size to 3 MB.
The DSPACE parameter in the VTAM startup limits the size for the VTAM data space that supports the VTAM Control Point application. It does not set a maximum for data spaces that VTAM gets for support of other applications.
It is the maximum allowed value of data space VTAM can take from the VSE system data space pool for itself.
The DSPACE parameter in the applications startup limits the value of data space which can be used by VTAM to support this specific application.

3. How VTAM uses Data Spaces

The default size allocated is 1 MB for each partition.
From the first 1 MB data space, 1/4 MB is used for data space management, the rest is also divided in 1/4 MB pieces and is reclaimed as needed.
There are two major considerations when trying to estimate the size needed for the VTAM data space:

1. The range of sizes of the request units (RU) being received by applications in the partition.
2. The number of RUs that have to be queued because the application does not have RECEIVE RPLs issued to read incoming RUs.

There are 10 possible different RU size groups. Each group will obtain a 1/4 MB piece when the first RU of this size is queued. The groups have following size limits (in hex):

Group	1	2	3	4	5	6	7	8	9	10
Size (upper limit)	78	F8	1F8	3F8	7F8	FF8	1FF8	7DF8	FBF8	10108

That means 3 MB of data space would allow to have RUs of all 10 groups - 1/4 MB would be left. A 2 MB data space would allow 7 groups, a 1 MB data space only could have 3 different groups. If one quarter MB is filled up with RUs not being processed, that means the RUs are not received, the related section will be extended by another 1/4 MB piece if further RUs need to be stored for this group. The first quarter MB reclaimed is not freed till VTAM is shut down, additionally used quarters for a certain group are freed if it is no longer used.
So the maximum size of the data space needed for VTAM and its applications is variable and VTAM extends dynamically.

4. Detailed Calculation

Minimum size used = 1 + N1 + N2 + N3 (MB)

where N1 is the number of VTAM applications using RUs with sizes of 3 different groups

N2 is the number of VTAM applications using RUs with sizes of 4 to 7 different groups

N3 is the number of VTAM applications using RUs with sizes of 7 to 10 different groups

The above minimum size is the minimum value that should be specified in the DSIZE operand of the SYSDEF command.

5. How to verify the settings

The total amount of storage available for data spaces is limited through the DSIZE parameter of the SYSDEF command. The standard environments A and C have a DSIZE value of 4 MB, environment B has 8 MB.

In case data compression is used, 2 MB of data space are need for the compression logic. This data space may extend by another 2 MB if needed. The DSIZE parameter should be increased accordingly if data compression is used.

The value of DSIZE should be the sum of all VTAM related data spaces, virtual disks and private data spaces used in the system.

We recommend to start with a higher DSIZE value than calculated and adjust the value according to the used space.

a) The QUERY DSPACE command will show the usage of data spaces:

```
                          DSIZE      MAX   PARTMAX      COMMAX   VDISK   DFSIZE
DEFINED:                  8192K      256       16           5       1     960K
ACTUAL:                   3072K        3        3           0       0

AREA  DSPS  AREA  DSPS  AREA  DSPS  AREA  DSPS  AREA  DSPS  AREA  DSPS
  F3     3
```

The ACTUAL: line shows the amount of space used (3072K), the last line shows how many data spaces are used by VTAM (F3).

b) The QUERY DSPACE,F3 command will show more details about the data spaces of VTAM:

```
AREA  DSPNAME          SIZE     MAXSIZE    SCOPE    OWNER  DU-AL  PASN-AL
  F2  ISTA95B8         1024K    2048K      ALL      F3             X

  F3  IST9A785         1024K    2048K      ALL      F3             X
  F3  ISTFB726         1024K    2048K      ALL      F3             X
  F3  ISTA95B8         1024K    2048K      ALL      F3             X

  F4  ISTFB726         1024K    2048K      ALL      F3             X
```

More info on VTAM 4.2 Data Spaces ...

6. In case a change is necessary:

If the maximum value of the data space size used by VTAM should be changed, the related EXEC statement has to be changed, e.g. for VTAM:

```
// EXEC ISTINCVT,SIZE=ISTINCVT,PARM='CUSTNO=...',DSPACE=4M
```

This change is in the VTAMSTRT - use skeleton SKVTAM. For the other VTAM applications proceed accordingly.

If the global value should be used, change the SYSDEF command in the ALLOC procedure using the related skeleton SKALLOCx where x is the environment character A, B or C.

NOTE:

Do not add the SYSDEF command into your IPL procedure, the subsequently called ALLOC procedure will deactivate your specification. If you need it in the IPL procedure e.g. if you use a virtual disk for the label area, be sure that you have the same values specified in the ALLOC procedure or that you have deleted the SYSDEF command there.

Ù Further info

Further VTAM 4.2 related performance info may be obtained from

„ 'VTAM V4R2 Release Guide', GC31-8090, Feb 95
Chapter 6: 'Performance Enhancements'

„ 'VTAM V4R2 Network Implementation Guide'
SC31-6494-00, Feb 95
Part 5: 'Tuning VTAM for your environment', 40 pages

„ 'VTAM V4R2 for MVS/ESA Performance Measurements'
WSC flash 9424, Doc-ID G023401, 06/17/94

„ 'Getting the most from APPC: A Network Tuning Guide'
John Brady, APPC Market Enablement, IBM Raleigh
Document for network administrators, March 94, 69 pages

„ VSE/VTAM Performance and Tuning Basics
by Jon vonWolfersdorf, IBM VSE/ESA System Center, Endicott
VM and VSE Tech Conf, Kansas City, 05/97, Session 33H
VM and VSE Tech Conf, Mainz Germany, 06/97, Session 53H

„ VSE/VTAM Tuning Basics
by Jon vonWolfersdorf, IBM VSE/ESA System Center, Endicott
WAVV 98 Conference, Albany, NY, 10/98

More info on VTAM 4.2 Data Spaces ...

Useful Commands for VTAM 4.2

- **GETVIS F3**

  shows usage of 24/31-bit GETVIS in VTAM partition F3

- **MAP F3**

  shows allocation information about VTAM partition F3

- **QUERY DSPACE,ALL**

  shows detailed information on VTAM's dataspaces

- **D NET,BFRUSE**

  shows information about all buffers and usage of 24/31-bit System GETVIS (optionally use ',BUFFER=SHORT')

- **D NET,VTAMOPTS**

  shows information about all startup options

- **D NET,STATS,TYPE=VTAM**

  shows information about the VTAM network

- **D NET,STORUSE,DSPNAME=***

  shows information about all dataspaces accessed by VTAM

- **D NET,STORUSE,APPL=appl-name**

  shows information about the dataspace accessed by VTAM for the specific application 'appl-name'

- **D NET,STORUSE**

  shows usage information about all VTAM storage pools

- **D NET,SESSIONS**

  displays session status information

PART H.

Languages and LE

High Level Assembler R2

Ù High Level Assembler R2 in VSE/ESA 2.1 Base

IBM High Level Assembler for MVS & VM & VSE Release 2 provides numerous enhancements in function, usability, performance, and reliability
... over all previous assemblers

Ù Performance Aspects

„ Virtual storage constraint relief

Utilization of storage above the 16M line

The assembler itself can be placed in 31-bit storage (430K in SVA-31).
Only a small I/O interface module must remain below (10K in SVA-24)

- Reduces need for scarce 24-bit storage
- Can also reduce/eliminate need for utility file I/O

„ Performance improvements

Vs High Level Assembler/MVS & VM & VSE Release 1:

- Generally improved performance characteristics
- Resource savings by utilizing more memory
- Processor usage for large macro-based assemblies has been reduced

Vs DOS/VSE (VSE/AF) Assembler:

- Performance is greatly improved

Vs Assembler H:

- In most situations fewer processor and I/O resources required
- For large and heavily macro-based assemblies more processor cycles may be used
- at assembly-time no exit to de-edit E-decks required (in VSE/ESA 2.1 all VSE macros will be in A-deck form)

High Level Assembler R2 (cont'd)

Ù Factors Improving Performance

This information has been compiled from Chapter 8 of 'High Level Assembler for MVS & VM & VSE: General Information', GC26-8261

„ Resident Tables and Source Text

Keep - intermediate text
- macro definition text
- dictionaries
- symbol tables

whenever possible in virtual/real storage.

Use workfiles on disk only when working storage is exhausted

„ Shared Virtual Storage

440K = 10 ASM phases SVA-31	eligible
11K = 2 ASM phases SVA-24	only

Reduces virtual/real requirements in a multi-user environment, and avoids FETCH I/Os

„ 31-bit Addressing

Concept of 'Data In Memory': More in-storage assemblies

The following areas may reside above the line:
- most of ASM routines
- work areas

Areas which must stay below the line:
- I/O areas
- exit parameter lists
- few initialization routines

„ Multiple Assembly

'Batching' multiple (related) assemblies reduces JCL overhead

„ Consolidating Source Text Passes

Ù For more info refer e.g. to 5 HLASM presentations by John R.Ehrman, SHARE 88, 03/97, as 88s288xh.pdf files in the Internet or to 'HLASM', by J.Dravenieks and J. Alexander, WAVV 97, 11/97

LE/VSE Overview on Charts

- **LE/VSE Relative Performance**

- **LE/VSE CPU-time Hints for Execution**

  General

  EXEC CICS LINK

- **LE/VSE Space Optimization**

  Compiler Options

  Run-Time Options

- **LE/VSE and the SVA (VLA)**

LE/VSE General Aspects

Overview

„ LE/VSE brings a lot of new language functions

especially for PL/I, apart from 31-bit

Better program isolation

„ Compile performance often is of minor interest

Often compiles are done on a workstation

Programs compiled on host are expected to be mostly already running ones, not newly developed ones

Compilation done much less often than execution

„ Excecution performance is of major interest

Run-time (Elapsed time) and CPU-time plus number of I/Os

COBOL, PL/I, and C

Batch and CICS environment, to be distinguished

LE allows tuning of runtime options on a program by program base

Some General Execution Aspects

„ Increased scope of language functions

When LE/370 was introduced on S/390, some increased CPU-time was seen due to that reason

„ COBOL/LE initialization

This time is slightly higher than COBOL II

-> percentage of CPU-time increase may be higher, especially for short batch executions or frequent EXEC CICS LINKs (even with all PTFs installed shown here)

Cont'd on next page

LE/VSE Relative Performance Aspects

Some General Execution Aspects (cont'd)

„ Batch OPEN/CLOSE

COBOL DOS/VS had OPEN/CLOSE routines inside of the batch program,
COBOL II and COBOL/LE dynamically build DTFs:
means slight extra work

„ LE performance benefit items

In certain cases, performance degradation may be even compensated:

- PL/I was in VSE an elder level (Some PL/I performance improvements were not included in VSE before LE/VSE)

- Higher buffer sizes by the ability to reside above the line (type of DIM exploitation)

„ Rough estimates for Batch

about 5% lower ITR vs non-LE (very dependent)

„ Observations for CICS/COBOL Environments

ITR-ratios (and TD NPS values)			
Workload	DOS/VS COBOL	VS COBOL II	LE 1.4 * COBOL/VSE
DSW	1.0 (0.265)	0.95e(0.26e)	0.95 (0.276)
RAMP-C	1.0 (0.308)	0.98e(0.30e)	0.98 (0.359)
Worst case **	-	1.0	- 0.90 -

e Estimated value
- Values asssume the same VSE/ESA environment (release and other deltas NOT included)
- All compiles are assumed with a 'comparable' OPTIMIZE

* LE/1.4 with performance PTFs for EXEC CICS LINK
** Worst case is a heavily 'subroutinized' customers load using EXEC CICS LINK

- 'Your mileage may vary'

	LE/VSE 1.1	LE/VSE 1.4
Based on	LE/370 1.2	LE/370 1.4
Applies to VSE/ESA	1.4.0-1.4.2 2.1.0-2.1.3	1.4.3 2.2.0 & up

„ DOS/VS COBOL was withdrawn from marketing 03/97.
To migrate it from old systems, see APAR PQ00970

CPU-Time Performance Hints (at run-time)

„ DFSORT STXIT/NOSTXIT option

Use NOSTXIT or MINSTXIT, if possible. This is especially beneficial, when TRAP(ON) is used as LE run-time option.
Refer to separate DFSORT/VSE document 'Program Invoked SORT'

„ Reduced CPU-time for COBOL/VSE jobs accessing VSAM (APAR PN84947,PTF UN92489)

Refer to the APAR/PTF section in this document.

„ Compiler options, improving run-time performance

The following options are optimal in case of COBOL/LE (if they can be selected function-wise):

NODYNAM	NORENT
FASTSRT	NOSSRANGE
NUMPROC(PFD)	NOTEST
OPTIMIZE	TRUNC(OPT)

Use the 'reverse' options only for debugging, not for production.

TRUNC(BIN) should be used only for selected programs that require guarantted non-truncation of binary data.
Refer e.g. to the COBOL/VSE Programming Guide.

Ask your IBM representative for a copy of the COBPERF package (many of the recommendations there are also valid for LE/VSE)

CPU-Time Hints (cont'd)

LE/VSE Run-Time Options:

„ Use ALL31(ON) also for Batch (vs ALL31(OFF))

Also HEAP and STACK storage may reside above the line

í Change the installation default for batch to ALL31(ON),

if AMODE24 is no longer required for legacy batch programs

With All31(ON)

- there is a very minor reduction of pathlength,
  since no code is processed to do mode switching.

- batch programs are no more bound in size.

ALL31(OFF) (plus STACK(BELOW...)) is required whenever an application in AMODE(31) CALLs an AMODE(24) program.

Note that the CICS default is already ALL31(ON).
Refer also to the ALL31 description on a following chart.

„ Specify enough working storage for LE pgms

Runtime options HEAP and STACK in CEECOPT for CICS may not be defined big enough (default is 4K, up to 20K were observed).

Use the RPTSTG option temporarily for an application in order to monitor its space requirements.

Increase these values if more workspace is required than defined, in order to avoid additional CICS GETMAINs and thus also VSE GETVISes.

In very specific cases, use the CICS/VSE Auxiliary trace:
- Start it for up to 10 sec
- Look for addt'l 'SCP ACQUIRED EXTENDED STORAGE' entries

CPU-Time Hints (cont'd)

„ CBLPSHPOP option

Check the setting of this option. If ON, it avoids compatibility problems but costs some performance (space, CPU-time) by issuing CICS PUSH HANDLE and CICS POP HANDLE commands every time a COBOL/VSE or COBOL II subprogram is called.

Use CBLPshpop(ON,OVR),

if your COBOL subprograms do not contain CICS CONDITION, AID, or ABEND condition handling commands.

„ RTEREUS option

Specifies a reusable LE environment for COBOL programs. Keeps initialized reusable COBOL/LE environment and thus may save initialization pathlength.

Not recommended in general due to restrictions with multienclave or multilanguage applications.

May be useful for non-LE programs calling repeatedly a COBOL/LE batch program.

Should only be used as part of CEEUOPT.

„ TRAP option

Allows LE to intercept an abend, provide abend info and then terminate.
TRAP(ON) is more-or-less a required option, BUT has some overhead, if a called program uses STXIT calls by its own.

CPU-time Hints (cont'd)

LE/VSE Run-Time Options (cont'd)

„ Check the use of the CHECK run-time option

If your COBOL program was compiled with SSRANGE and you are not testing or debugging an application ...

Save CPU-time by setting CHECK(OFF)

When program is debugged, you should/can re-compile w/o SSRANGE, which may give a minor gain in CPU-time.

Applies to COBOL only

„ Check the use of the DEBUG run-time option

Activates the COBOL batch debugging features specified by the USE FOR DEBUGGING declarative.

Save CPU-time by setting DEBUG(OFF).

Use DEBUG(ON) only during application development or debugging.

Applies to COBOL only

í APAR PQ15901 (LE), PTF UQ17824 (98-05-19) changes CHECK and DEBUG

in CEECOPT, CEEDOPT, and CEEUOPT

plus STORAGE heap_alloc_value from NONE to 00.

CPU-Time Hints and EXEC CICS LINK

„ Use Dynamic CALL instead of CICS LINK, if possible

EXEC CICS LINK has higher CPU-time overhead since it causes LE/VSE to create a nested enclave with its own resources, such as storage (GETVIS/FREEVIS).

Dynamic CALL is not 'PPT driven', so the CALLed routine may not reside in another CICS partition (as for DPL)

- EXEC CICS LINK may show a bigger impact if applications are 'subroutinized' heavily

Unfortunately, CALL could not be used in general for DOS/VS COBOL, so most existing source contains EXEC CICS LINK.

If you change source code, make sure that no CALLed program tries to return via EXEC CICS RETURN.

„ GETVIS Performance PTF for EXEC CICS LINK

Install the following CICS/VSE 2.3 performance PTFs:

UN99292 and UN99294 (APAR PN91969)

This code improves the LE enclave creation, by more effectively using VSE GETVIS/FREEVIS, together with a separate GETVIS subpool.

The CPU-time benefit for production workloads, naturally, depends on the relative frequency of enclave creations (i.e. mostly of EXEC CICS LINKs).

Runs with the RAMP-C Online workload showed about 10% less CPU-time per tx (without the use of LESTG described below).

Even higher benefit has been observed by the CICS PTF, described on the next foil.

„ Newly provided PTFs

Refer to 'Recent/Forthcoming LE Perf. PTFs'.

CPU-Time Hints and EXEC CICS LINK (cont'd)

„ CICS/VSE LESTG PTF for EXEC CICS LINK

Install the following CICS/VSE 2.3 performance PTFs:

UQ06448 and UQ06449 (APAR PQ03907)

This PTF (even more than PN91969) avoids too high 31-bit LE-txn overhead for the EXEC CICS LINK enclave creation. CICS/VSE internally manages the dynamic storage allocation in CICS owned storage above the line. This storage above the line is only available for LE (1.1 and 1.4) program initialization.

To exploit it, a SIT or SIT overwrite option is required:

LESTG= nnnn

nnnn is in K, with 256 as minimum and 65536 as maximum value. Default is NO.

In case of LE 1.4, make sure before using LESTG, you have the following PTFs applied (to avoid DFH0506 SOS below the line due to an incorrect control block initialization):

UQ10085	LE/VSE 1.4 prereq
UQ10085	LE/VSE 1.4
UQ12342	CICS/VSE 2.3
UQ12343	CICS/VSE 2.3

New CICS Shutdown statistic counters for LESTG, show

- Maximum usable LESTG area
- Maximum used LESTG (a high water mark)
- # successful LESTG GETMAINs
- # unsuccessful LESTG GETMAINs (i.e. how often the LESTG area was exhausted and thus VSE GETVIS requests were done, enhanced by PN91969).

They are also included in the display of the CSTT transaction.

„ Recent LE/VSE APAR PQ23382

See separate foil

LE/VSE Space Optimization

Ù **LE/VSE space optimization is beneficial to**

- **reduce program compressions in CICS**
- **allow larger batch programs (bring data in from disk)**

Note: Apply recent LE/VSE APAR PQ23382 to get big VSCR

Refer to separate foil

Storage Areas for LE/VSE

Area name	Content	Location given by
Application Program Working Storage	User programs Data items for applic.	Resulting RMODE HEAP, DATA, RENT	
External Data	Area to pass data	ALL31, DATA, RENT	
Misc. LE Internal Areas	Automatic variables, 'HEAPs' Library rout.variables 'HEAPs' I/O control blocks , 'HEAPs' Library stack storage, 'STACKs' Save areas 'STACKs'	
Misc. LE Control Blocks	About 14K for Batch, about 61K for CICS	Always below	

- For 'HEAP' and 'STACK' storage, refer to LE/VSE Run-Time options

Options Affecting LE/VSE Storage

Ù COBOL LE/VSE Compiler Options

DATA(24!31)	Where to put pgm working storage for RENT, and external working storage. (DATA(24) required for DFHDRP use)

RENT!NORENT	RENT causes reentrant code (SVA eligible), to be shared among multiple partitions (RENT is required for CICS programs). NORENT causes program working storage to be part of the application phase.

RMODE(...)	For RENT programs: Controls where appl. may reside. For NORENT programs: Controls where the working storage resides (together with the application)

	RENT programs	NORENT programs	
RMODE(24)	24	24	
RMODE(ANY)	ANY	ANY	
RMODE(AUTO)	ANY	24	(default)

Ù Linkedit Options

AMODE=24!31!ANY	What addresses a program uses and understands
RMODE=24!ANY	Where a program is loaded (RMODE=24 required for DOS/VS COBOL)

í 3 valid combinations for execution:

- AMODE24/RMODE24 - AMODE31/RMODE24 - AMODE31/RMODEANY

í Try to achieve that your programs are cataloged with AMODE 31 and RMODE ANY.

Understand why, if not. Remove reason if possible.

Options Affecting LE/VSE Storage (cont'd)

Ù LE/VSE Run-Time Options

ALL31=(ON!OFF) Specifies	whether or not an application can totally run in AMODE=31. ALL31(OFF) together with STACK(,,BELOW) is required for an application with an AMODE=24 pgm: - a COBOL II NORES pgm - a DOS/VS COBOL pgm (batch) - a non-COBOL AMODE(24) pgm) ALL31(OFF) causes LE - to setup and perform AMODE switching on calls to LE services - to allocate storage for the AMODE(24) LIBVEC (library routine vector) (about 22K) below - to allocate run-unit control blocks (about 8K) below

See also LE APAR PQ23382, regarding 'AMODE24 autodetection'.

'Heap Storage' are areas shared among all program units and threads in an enclave, so lifetime is not related to the execution of the current routine:

ANYHEAP	Size of ANY storage for library routine variables (about 4K/12K for small/big programs).
BELOWHEAP	Size of BELOW storage for library heap storage and control blocks for I/O (LE internal structures).
HEAP(,,ANY!BELOW) Size of	initial and additional heaps, used for user controlled dynamically allocated variables . Use 20K/50K for primary/secondary allocation.

Options Affecting LE/VSE Storage ...

'Stack Storage' is allocated at entry to a routine or block, and freed upon return:

LIBSTACK	Size of BELOW save areas for library stacks (used only by library routines).
STACK(,,BELOW!ANY)	Controls the allocation of thread's stack storage (used by library routines and compiled code). STACK(,,BELOW) required for ALL31(OFF).

STORAGE(,,,xK)	xK defines the amount of storage, LE reserves to handle out-of-storage conditions. This area is always below the line.

„ Check the STORAGE option

in CEEDOPT and CEECOPT for Batch and CICS (or CEEUOPT).

Be aware that NONE does not initialize working storage.

Application Pgm Control of Areas

Ù Applic. Pgm Control of Data Location

Options for an application			Working Storage Control			
Run-Time	Compile-Time	Blocks	Non-(program)	External	Appli-cation	
ALL31(ON)	RENT	DATA(31) X1	X	X	X	*5
*1		in ANYHEAP	in ANYHEAP	in HEAP	in HEAP	
		DATA(24) X1	-	-	X	*5
*2		in ANYHEAP	in ANYHEAP	in HEAP	in HEAP	
ALL31(ON)	NORENT	DATA(31) X1	X	X	X	*5
		*4 in ANYHEAP	in ANYHEAP	in app	in HEAP	
		DATA(24) X1 in ANYHEAP	X in app	- in HEAP	-	
ALL31(OFF)	RENT	DATA(31) x2	X	-	X	*5
		in ...HEAP	in HEAP	in HEAP		
		DATA(24) x2 in ...HEAP	- in HEAP	- in HEAP	-	
ALL31(OFF)	NORENT	DATA(31) x2	-	-	X	*5
		*4 in ...HEAP	in app	in HEAP		
		DATA(24) x2	-	-	-	
*3		in ...HEAP	in app	in HEAP		

X Eligible for residing above the line

X1 As many as possible
x2 Some control blocks in ANYHEAP, some in BELOWHEAP

*1 Minimizes space requirements below the 16M line
*2 All working storage BELOW
*3 Maximum requirements for BELOW
*4 NORENT not allowed for CICS
*5 Location depends on RMODE or resulting RMODE

External Working Storage resides above the line,

if - HEAP(ANY) is in effect, plus
- DATA(31)
- plus ALL31(ON)

Hints for LE/VSE Space Optimization

Hints for LE/VSE Compiler Options

- **Try to use RENT and DATA(31)**
- **Specify DATA(24) only for programs passing data to programs in AMODE=24**

Hints for LE/VSE Run-Time Options

- **Try to run any application with**

  ALL31(ON) and HEAP(,,ANYWHERE), STACK(,,ANYWHERE)

  ALL31(ON) needs AMODE=31 for all programs in the run unit.

- **Specify sufficient (but not too much) initial sizes for ANYHEAP, BELOWHEAP, HEAP, LIBSTACK, STACK**

  to avoid too frequent CICS GETMAINs
  (refer to 'CPU-time Hints' chart)

- **Use the RPTSTG option (only initially),**

  to check space requirements of an application
  (e.g. as PARM parameter in the batch EXEC statement)

- **Use the RPTOPTS(ON) option (only initially),**

  to display currently valid default options

- **Do not set 'harmful' options in CEEDOPT (Batch) or CEECOPT (CICS)**

  Use CEEUOPT instead to make them application specific,
  or use EXEC .., PARM= '..' for individual batch programs.

Virtual Storage Hints

Refer especially to

'Exploiting 31-bit Capabilities with COBOL/VSE and LE/VSE',
by J. Winchell, VM and VSE Tech Conf 05/97, Kansas City
VM and VSE Tech Conf 05/98, Reno Nevada, Session 31I

„ 31-bit applications with PL/I available

This gives VSCR for other applications in the same CICS partition

„ Make sure your LE programs are 31-bit eligible

Check the link lists and look for 'AMODE RMODE' = '31 ANY'

„ Ensure that RMODE(ANY) really applies for 31-bit COBOL/LE programs

If defaults not settable, use the CBL option RMODE(ANY)

„ More virtual storage required

The LE environment in general requires more virtual storage
(e.g. COBOL/LE vs DOS/VS COBOL).

Fortunately the increased storage is predominantly above the
line (CICS GETMAINs are converted to VSE GETVIS ANY requests).

Storage tuning requires knowing your application and making
experiments in your specific environment (see the RPTSTG option)

AMODE(31) and RMODE(ANY) do not increase the size of LE programs.

More LE/VSE Performance Hints ...

Virtual Storage Hints (cont'd)

„ Use EXEC xxxx, SIZE=AUTO for batch PL/1 pgms

Otherwise there is no use of storage between the end of the load phase and the begin of the GETVIS storage

„ Avoid GETVIS-24 orphans in CICS

- Refer to Information APAR II06138 ('CICS reserved verbs')
- Always use the WORD=CICS compile option for LE-programs to run under CICS
- Follow the application programming rules in the 'CICS Application Programming Guide' SC33-0712-02 p17 ff

General Hints

„ Understand 'LE/VSE Performance Considerations'

for

AIXBLD	ALL31	ANYHEAP	BELOWHEAP
CHECK	DEBUG	HEAP	LIBSTACK
RPTOPTS	RPTSTG	RTEREUS	STACK
STORAGE	TEST	TRACE	TRAP

in Chapter 2 'LE/VSE Run-Time Options' of the LE/VSE Programming Reference

LE/VSE Phases and SVA (VLA)

„ General

- There is no functional requirement to load any LE phase into the SVA (VLA), except CEEEV003 and EDCZ24 (see below)

- Loading any phase into VLA may save
  - LOAD I/Os
  - VSIZE, if many partitions use LE concurrently

- Shipped LE load books include both 24 and 31 bit phases (they would force you to load both types)

- Refer to 'Space Optimization' for general SVA/VLA hints

„ VLA-24 for LE

Loading SVA-24 phases for LE in general is NOT recommended before APAR PQ23382.

- The following SVA-24 eligible LE Base phases are only loaded once per job step
  (not relevant for CICS, only for short batch job steps/pgms):

	Old	New	
CEEBINIT	294K	53K	LE batch
CEECCICS	300K	47K	LE CICS (in CICS PPT)
CEEPIPI	305K	65K	LE pre-initialization
CEEPLPKD		74K	LE DTF builders (mainly)

Do NOT load the old versions of these phases into SVA-24.
Apply the PTF for APAR PQ23382 to get the new ones, and put them into the SVA-24.

But, since they are 24-bit, their old sizes may have been a problem already regarding CICS private space below the line.

„ VLA-31 for LE

Load the following LE runtime phases into VLA-31:
They are required once per LE-program execution, and thus of special benefit for Batch (non-CICS) use:

CEEEV005	15K	COBOL for VSE/ESA	
CEEEV010	186K	PL/I	-"-
CEEEV003	1186K	C	-"- (Req'd for CICS)
EDCZ24	1266K	C	-"- (Req'd for Debug Tool in CICS)
CEEPLPKA	302K	LE	-"- New, offload from SVA-24

Make PSIZE-31 big enough to avoid 'downloading' them into SVA-24

Recent LE/VSE Perf. PTFs

The following LE related PTFs are being provided, based on
- Recent LE/VSE enhancements
- SYSROUTEd PTFs from LE OS/390
- CICS/ESA PTFs in OS/390, for CICS TS in VSE/ESA 2.4

They result in
a) big VSCR for LE environments
b) less CPU-time for EXEC CICS LINK

Ù APAR PQ23382 (LE), PTF UQ27971

a) 'VSCR for LE/VSE' part

LE/VSE code was reduced below the line by moving many routines out of the following SVA-24 eligible initialization phases:

SVA-24-el. phase	Old size	New size	Relief	
CEEBINIT	294K	53K	241K	
CEECCICS	300K	47K	253K	
CEEPIPI	305K	65K	240K	
Sum	899K	165K	734K	

- Phase sizes varied slightly before this APAR (depending on service level)
- Moved routines now in SVA-31 phase CEEPLPKA, and consolidated to fewer ones. CEEPLPKA increased by 244K (from 58K to 302K)

Í Significant VSCR for all LE/VSE uses

- All environments (Batch, CICS/VSE, CICS TS)
- All languages

Now, these phases should be loaded into SVA-24 in any case.
Refer also to chart: 'LE/VSE phases and SVA(VLA)'.

Also a sysroute of an LE OS/390 APAR is included:

- PQ178931 (LE) Allows ALL31(ON) as default in CICS partitions, by dynamically switching an application to ALL31(OFF), if required:
'AMODE24 autodetection' for statically linked pgms

Cont'd on next page

Recent LE/VSE Perf. PTFs (cont'd)

Ù APAR PQ23382 (cont'd)

b) 'EXEC CICS LINK' part and COBOL 'Skip Exit DSA processing'

The following LE OS/390 related APARs are included:

- PQ14883 (LE) Reduced EXEC CICS LINK pathlength by fewer GETMAINs (up to 10% for ALL31(ON), 25% for ALL31(OFF)). Also allows languages to skip Exit DSA processing. Some other optimizations in the Init/Termination path of enclaves/threads (CICS/VSE and CICS TS).

- PQ14888 (LE) PQ22514 Reduced EXEC CICS LINK pathlength by reuse of areas for ALL31(ON)(up to 15% reduction, CICS TS). Also some other non-CICS LINK path reductions.

Performance improvement (reduced CPU-time) for

- Batch (some), CICS/VSE, especially CICS TS
- All languages, especially COBOL/VSE

Ù APAR PQ23385 (COBOL/LE), PTF UQ28062

- PQ16794 (COBOL) Enable Skip Exit DSA processing for COBOL/VSE. (Required on top of APAR PQ23382)

Ù APAR PQ16844 (OS/390), code std in CICS TS

- PQ16844 (CICS OS) Enables the reuse of areas provided by PQ14888. CICS TS keeps track of RUWA storage to minimize GETMAINs. Comes along with a new CICS TS SIT parameter 'RUWAPOOL=NO!YES'. No is the default and the old method w/o reuse.

Ù More info

Refer to: 'LE/VSE and DT/VSE- Performance Improvements and Support for CICS TS' by Peter Van Dyke, IBM Australia. VSE/ESA Software Newsletter (forthcoming, 2Q99)

IBM Debug Tool for LE

Needs LE/VSE 1.4 (or the C Language Run-Time Support)

„ Function

Examine and monitor execution of LE programs:

COBOL, PL/I, and C

Insert hooks into the programs via the compiler TEST options

Invoke Debug Tool via the run-time TEST options

„ Performance Impact

Higher CPU-time overhead if used

A lot of extra activities is being done,
overhead is high, but adequate to the added function

Some CPU-time overhead, even if not used

Around 5% to 10% more CPU-time by additional hooks in the compiled code.

Also, increased program size

í Finally, do a compilation w/o compiler TEST option

„ More info

'IBM Debug Tool' -User's Guide and Reference-, SC26-8797-00, 08/96

LE/VSE References

For more info refer e.g. to

- 31-bit Addressing in PL/1 for VSE, Getting Started,
  ITSO Red Book GG24-4271-00, 01/95, 83 pages

- IBM COBOL for MVS and VM Version 1.2:
  'Performance Tuning', 01/96, 44 pages
  COBPERF package on MKTTTOLS, also available via Internet
  http://www.software.ibm.com/ad/cobol/cobol.html

- MVS Performance: 'Potential Performance Improvements for COBOL
  Programs Running on CMOS Processors',
  WSC Flash 9608.2, 02/96

- IBM LE for VSE 'Installation and Customization Guide',
  SC33-6682-01, 08/98

- IBM LE for VSE 'Programming Reference', SC33-6685-0x, 12/96

- IBM LE for VSE 'Programming Guide', SC33-6684-0x, 12/96

- 'Taking Advantage of LE/VSE', SG24-4798-00
  ITSO Boeblingen Red Book, 10/96, 85 pages
  Appendix A4: Performance Recommendations

- 'IBM COBOL and Language Environment for VSE
  - How to upgrade now -', SG24-4277-00,
  ITSO Boeblingen Red Book, 05/97, 150 pages

- 'Exploiting 31-bit Capabilities with COBOL/VSE and LE/VSE'
  By Janice Winchell, VM and VSE Tech Conf
  05/13-16/97, Kansas MO, Session 31C

- 'It's Time to Upgrade to COBOL for VSE/ESA'
  By Alice Crema, VM and VSE Tech Conferences,
  06/97, Mainz, Germany, Session 51A
  05/98, Reno, Nevada, Session 31B

- The LE for VSE Homepage on the Internet:
  http://www.s390.ibm.com/le_vse/

PART I.

Misc. VSE/ESA Products

Client/Server Performance Aspects

- C/S enhancements may cause different load on VSE

  LANRES and ADSM support enable additional functions for a VSE host

  - increased load/exploitation of S/390 H/W

  VisualGen (e.g. for VSE IUI/workplace) does not put load on VSE

  VisualGen generated applications provide improved execution performance (production environment)

  COBOL vs interpretative code
  CPU-time, I/Os (?), Virtual storage (?)

  CICS/APPC (LU6.2) support gives improved DWF response times

LANRES/VSE

Allow a NetWare or OS/2 server to exploit S/390 resources

Ù LANRES/VSE functions

- Disk Serving
- Print Serving
- Data Distribution
- LAN Administration

Ù LANRES performance PTF

LANRES/VSE 6.1.1 PTF UQ03000 (APAR PQ01885) or newer

Uses up to 15 VSAM LSR subpools (1 for each drive letter) with at most 100 8K buffers for the Disk Serving function (Avoids higher VSAM CPU-time when partition is huge)

Ù Potential VTAM IOBUF exhaustion

When using the LAN to Host printing function for big files, the shipped defaults for the pacing parameters (0, no pacing) may finally lead to IST154I 'Expansion failed for I/O buffer pool'.

Problem can be avoided if a Logmode is used with pacing (Both the #BATCH and the #INTER Logmode have no pacing). Modify e.g. the #BATCH, as suggested below:

- Copy the logmode #BATCH from ISTINCLM to a private MODETAB, newly defined in the host application major node (APPL) and on the resource logical unit (LU)

- Change the 3 pacing values SSNDPAC SRCVPAC and PSNDPAC from X'00' to a nonzero value (e.g. X'02')

The Logmode #INTER should not be used for huge file transfers concurrent to other LANRES activities.

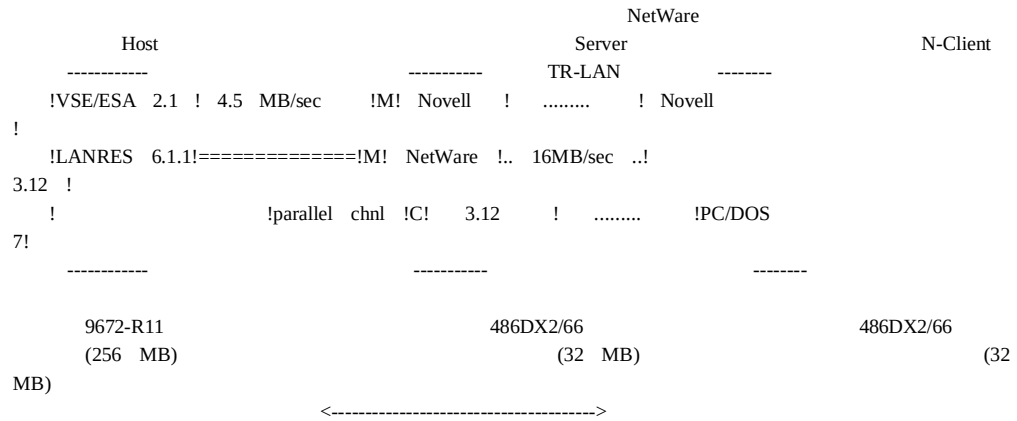
You may also refer to Doc-APAR PQ00467

Ù For the OSA-2 adapter, specify DELAY=0 for LANRES. The default DELAY=0.05 is not small enough for high speed transfer.

Ù Specify VPACING=8 in the VTAM APPL statement for LANRES. APPC users should be aware, VPACING currently is shipped as 4.

LANRES/VSE Performance

Ù Test Environment



Ù Benchmarks

„ Disk serving functions (file copy to/from host)

- 20 MB total amount of data
- VSAM RRDS file on 9345-1 volume, 8K CIs, LRECL=512 byte
- 1 LSR subpool in VSAM, catching all GETVIS
  (Partition size 5M and 30M resulted in same EDRs)
- Many small files (2000 files, 10K each)
  or some bigger files (20 files, 1M each)

Ù Observed Effective Data Rates (EDRs)

Function		EDR
10K files	Upload to host	159 KB/sec
	Download	61 KB/sec
	(Download to NUL)	(226 KB/sec)
1M files	Upload to host	485 KB/sec
	Download	393 KB/sec
	(Download to NUL)	(501 KB/sec)

í Good effective data rates

- WRITE to host faster than READ
  (or PC-READ faster than PC-WRITE)

- Download values to NUL indicate the PC-WRITE-time impact

- About 5M LANRES/VSE partition size recommended

SQL/DS 3.5 Performance Improvements

Optional product, included in VSE/ESA 2.1.2 and up

„ New Data Restore Feature

Selective log recovery can execute SQL/DS statements against RELOADed tables

Selectively RELOAD tables from an archive

í Allows continuous operation and improved system availability

„ Archive Performance Enhancements

- Only allocated pages of database and of log are archived
- Usage of Multiple-Block BLOCKIO requests (up to 28K) in a single IUCV SEND request (VM only)
- Usage of asynchronous I/O for concurrent disk and tape operations via double buffering (VM only)
- Usage of VSAM controlled buffers (instead of user buffering) exploits VSAM READ ahead for sequential access

í Archive in SQL/DS 3.5 significantly faster

(Improvements observed up to a factor of 5.7)

„ CICS Database Switching

Eliminates the need to stop and restart the resource adapter in CICS.

Allows concurrent access to multiple servers from a single CICS: single transactions can connect to different SQL/DS servers in different LUWs

í Facilitates distributed computing

SQL/DS 3.5 Performance Improvements (cont'd)

„ **Improved SHOW CONNECT performance display for DB administrators**

„ **Increased Buffer Maximums**

Up to 400,000 buffers for both directory and data,
larger than most customers will be able to use (1.8 GB)

	SQL/DS 3.4	SQL/DS 3.5
NDIRBUF (.5K direct.blocks)	28,000	400,000
NPAGBUF (4K data pages)	3,500	400,000

í **Keep more data in storage, reduce number of I/Os**

(requires corresponding amount of real storage)

HINT: Benefits of more buffers is limited by the need to write all modified pages to DASD at checkpoints

„ **Utilizing Virtual Disks for Internal Dbspaces**

Internal dbspaces are used as temporary workspace,
thus no risk to use Virtual Disks.
Best: Use 1 big Virtual Disk

í **Faster index creation, joins, sorts**

(provided virtual storage is backed by enough real storage)

(observed were Elapsed time reductions of up to 40%,
depending on the size of the index, 20 to 80 byte)

„ **Provision of**
- **Shutdown Statistics (OFF|ON)**
- **Checkpoint Statistics (OFF|ON|DETAILED)**

ARI... Modules and 31-bit Appls

Modules linked for COBOL programs with SQL/DS:

ARIPADR req'd for COBOL programs pre-processed prior to SQL/DS 2.2
(is AMODE/RMODE 24/24 for backward compatibility)

ARIPADR4 required for ANY COBOL program (incl. COBOL II, and
..... 5 COBOL/LE) pre-processed with SQL/DS 2.2 and up
(is AMODE/RMODE 24/24 before and ANY/ANY in DB2 for VSE
& VM 5.1.0)

ARISSMA required for ANY SQL/DS application if function is used

In order to place your SQL/DS COBOL programs above the line ...

- **Migrate to DB2 Server for VSE & VM 5.1.0**

or

- **Use the Linkage Editor statement**

  MODE AMODE(ANY),RMODE(ANY)

  ... to force AMODE/RMODE ANY.

  (Note that ARIPADR4/5 will work above the line before 5.1,
  even though defaulting to AMODE/RMODE 24/24)

or

- **Install the PTF for APAR PN91267**

  Makes the ARIPADR, ARIPADR4, ARIPADR5, and ARISSMA 31-bit
  capable, as is standard in DB2 Server 5.1

SQL/DS (DB2) Performance Information

For info/more info on SQL/DS performance, refer e.g. to

SQL/DS 3.4:

- 'VSE/ESA 1.3/1.4 Performance Considerations'.

SQL/DS 3.5 (VSE &VM):

- SQL/DS 3.5 Usage Guide, SG24-4647-00, ITSO Boeblingen Red Book, 12/95, 221 pages (See Chapter 3: 'Performance Benefits')

- SQL/DS Performance Tuning Handbook, SH09-8111

- SQL/DS Archiving and Recovery Using the Data Restore Feature, SG24-4833-00, ITSO Boeblingen Red Book 09/96, 94 pages Especially refer to Chapter 3 'Performance Benefits'

- DB2 for VSE & VM -Performance and Tuning- VM/VSE Tech Conf 06/96, Orlando, by Terence Foulds, IBM, 78 foils

- 'Improve Your Data Availability with SQL/DS 3.5' VSE/ESA Software Newsletter, 1Q/2Q 97, p4-8 ftp://lscftp.pok.ibm.com/pub/vse/docs/vsenew14.pdf and (for IBMers) as VSENEW14 PACKAGE on IBMVSE tools disk

DB2 for VSE & VM 5.1:

- DB2 Server for VSE & VM: Perform. Tuning Handbook, SC09-2402-00

- DB2 for VSE & VM -Diagnosis Guide and Reference-, SC09-2408-00 Especially refer to Chapter 5 'Diagnosing Performance Problems'

- 'DB2 for VSE & VM, Performance and Tuning' - By Terence Foulds and Debbie Yu, WAVV 97, Chattanooga TN, November 7-11, 1997

- DB2 Performance Tuning on VSE and VM, ITSO Boeblingen Red Book, 05/98, SG24-5146-00 (229 pages)

MSHP Performance Aspects

MSHP Performance Basics

Ù MSHP History File as repository for S/W changes

„ Contains chained records (1984 bytes each), describing S/W changes (module exchanges, module internal changes)

„ Records are used for MSHP service, reading 1 record per I/O

„ MSHP file usually is 10% to 30% full, often up to 70%
i.e. when certain vendor programs applied local service:
(changing several hundreds of lines in a member, instead of replacing it, a type of 'misuse')

„ More I/Os may be required, if
- MSHP History File is scattered
- 254 MSHP buffers are not sufficient to contain all data

MSHP Performance Hints

Ù Do an MSHP RETRACE from time to time to find out '% of space used'

Ù Do a MSHP History File reorg

- at a functional problem
- when too many I/Os are required
(mostly at high percentage of space used,
after a FSU of many optional products 'new installs')

Use the following MSHP job (provided in ICCF lib 59):

```
// JOB REPHIST
// DLBL IJSYS02,'WORK.HIST.FILE'
// EXTENT SYS018,SYSWK1,1,0,1065,90
// ASSGN SYS018,DISK,VOL=SYSWK1,SHR
* CHECK THE DLBL. IT IS FOR THE WORK HISTORY FILE ON
* YOUR SYSTEM, HERE SET UP FOR 9345.
* WE SUGGEST TO BACKUP THE HISTORY FILE BEFORE THIS JOB
// PAUSE /// EXEC MSHP
CREATE HIST AUX
COPY HIST SYS AUX
CREATE HIST SYS
MERGE HIST AUX SYS
/*
/&
```

DFSORT/VSE Performance

This part has been updated for DFSORT/VSE 3.4 and was moved into a separate document:

'IBM DFSORT/VSE Performance Considerations'

For more info, you can also surf the INTERNET:

http://www.storage.ibm.com/storage/dfsortvse/

PART J.

Appendix A: VSE Space Optimization

Shared and Private Space Optimization

Shared Space Size Dependencies

Here, essentially the space below the 16M line is considered

„ Total Supervisor Size:

- Ù Base size of the supervisor
  Depends on the generation options incl. IODEV for 8 byte PUBs
- Ù Number of ADDed devices (around 256 byte/device)
- Ù IPL SYS BUFSIZE, CHANQ, NPARTS, NPGR, SDSIZE
  Refer to VSE/ESA IPL Procedure chart

„ SVA-24 size:

- Ù VPOOL window size

  A VPOOL size of 64K is usually sufficient,...

  except you need to have more than 16 concurrent POINTs to files in VIO.

  This can happen with specific vendor programs which think they need 'more-than-1-concurrent-POINT-per-file' tricks.

  This may happen also if you have say more than 8 concurrent CICS/VSE partitions

- Ù IPL SVA command:

  Definition of SDL, PSIZE and GETVIS-24

- Ù IPL SYS command:

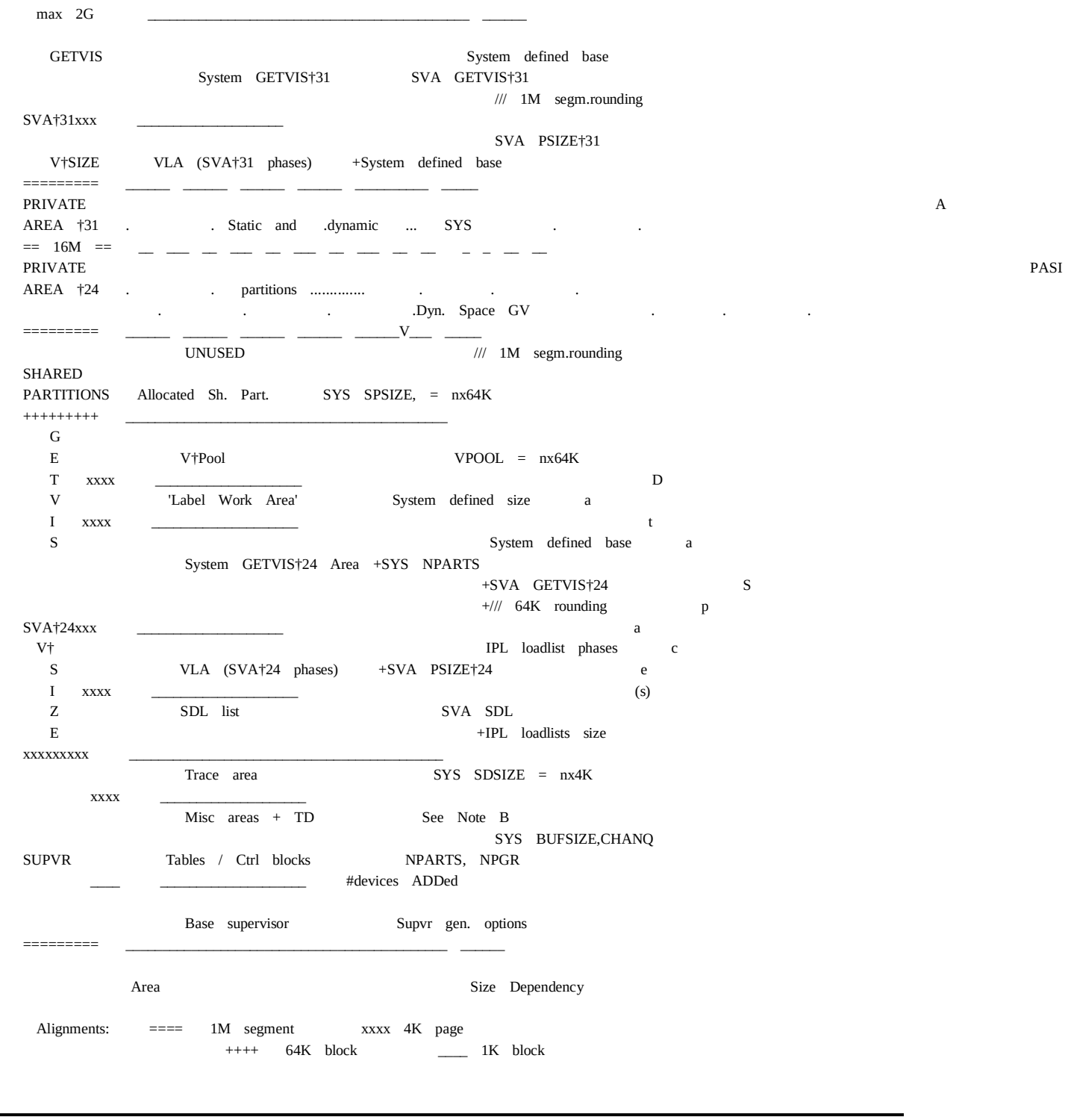
  NPARTS parameter for maximum number (NPARTS-12) of concurrently active dynamic partitions

- Ù Space needed for SVA resident phases in VLA (-24)
- Ù System defined requirements for System GETVIS-24

„ Size for Shared Partitions:

SYS SPSIZE=0 recommended for VSE/ESA V2,
except if shared partition required for vendor product(s)

VSE/ESA V2 Shared Space Layout



VSE/ESA V2 Shared Space Principal Layout (Notes)

Basic Principle: No holes in shared space

This chart shows the principal layout of the shared space areas for VSE/ESA V2.

In order to

Ù emphasize on VSE's consciousness of this precious resource

Ù facilitate VS tuning,

on the left hand side of the chart the units of alignment are shown. For the SVA, the subdivision into the SIZE and GETVIS parts, shown by the MAP command are given.

On the right half of the shared area, the relevant parameters are shown, determining the size of each individual area.

Shared space, which is required to achieve the 1 MB segment rounding is given to the shared partitions, even if SPSIZE=0K has been specified. This space for rounding is always a multiple of 64K, since space required to achieve 64K rounding is being given to the System GETVIS Area.

Note A:

The arrangement, sequence and layout of VSE/ESA shared areas have only been addressed above for illustration purposes. You should NOT use implicitly or explicitly any internal information in any coding for programs running under VSE/ESA, except the officially documented interfaces.

VSE/ESA shared storage layout may change, depending on technical needs.

Note B:

Turbo Dispatcher (TD) is loaded separately, if used.
TD size increased from initially about 25K to 34.7K (VSE/ESA 2.3, TD level 8)

VSE/ESA V2 Shared Space Layout (cont'd)

Notes/Remarks:

==== 1M segment boundary aligned

Shared space required for the 1M segment alignment
is as multiples of 64K added to SPSIZE (even for SPSIZE=0)
† in addition to the specified value, rounded up to nx64K

++++ 64K aligned

Shared space required for the 64K alignment
is as multiples of 4K given to System GETVIS Area
(Required since VSE partition sizes are multiples of 64K)

xxxx 4K page aligned

Shared space required for 4K page alignments is given to
† BUFSIZE (if within SUPVR area)
† System GETVIS†24 (if within SVA)

- **Any small delta in startup (specification, configuration) may cause that an area needs an increment for alignment.**

  Theoretically, 1 byte more may require a 64K or 1M rounding.

  BUT: Roundings can be compensated by VS tuning.

Hints for Determining Actual Sizes

„ MAP command

- Supervisor area size (includes SDSIZE)
- SVA-24 sizes:

  V-SIZE in SVA-24 = PGM AREA (VLA + SDL list)

  GETVIS in SVA-24 = VPOOL + SLA + GETVIS SVA

  UNUSED in SVA-24 = Specified SPSIZE rounded up to nx64K
  + Rounding for 1M segment (nx64K)
  - ALLOCations for shared partitions

„ IUI Display Storage Layout panels ('363')

DISPLAY STORAGE LAYOUT panel

Total size of shared space below the line:
Shared Partitions + SVA + Supervisor

SHARED VIRTUAL AREA LAYOUT panel

All individual sizes of the SVA-24 area

The actual SPSIZE (which includes UNUSED from MAP, and the ALLOCated shared partitions) can be determined by:

Total shared space-24 - 'SUPVR' - SVA-24

More Private Space

The following tuning hints for getting more private space in native mode and under VM can be applied basically in any order.

It is recommended, however, that "Check chances for gaining 1 MB segment more private space" is done after the other hints have been considered.

Ù General Allocation Rules

In order to avoid (sometimes unbeneficial) upward roundings of virtual size allocations, specify sizes in multiples of

4K (SIZE, ALLOCR)
or of 64K (VPOOL, VIO, ALLOC).

Ù Appropriate SIZE values

Specify whenever possible SIZE=AUTO in the EXEC statement.

This causes that a value is taken, which is needed to load the biggest phase with the same first 4 characters of the phase in the EXEC statement.

SIZE=AUTO cannot be taken for those programs, which dynamically expand the program storage into dynamic storage, without using GETVIS requests.
Such programs are e.g. the Linkage Editor, Librarian and Compilers.

Use GETVIS command to display the remaining GETVIS size.

Ù Dynamic Partition Space GETVIS is only used by specific pgms

In general, more than the default sizes shipped in the Dynamic Class Table cannot be exploited.
Refer to 'Dynamic Vs Static Partitions'

Ù Leave POWER in private partition

There is only a very marginal performance (CPU-time) delta vs POWER in shared space.

You would require up to 1 MB more shared space below the line.

More Private Space (cont'd)

Ù Leave VPOOL=64K in the IPL procedure

It was shown that it is not expensive to 'connect a VIO-page to a page in the smaller VPOOL window'.
(See also the POINT remarks on a previous foil).

VIO requirements:

Linkage Editor	64K
Primary CICS (DBDCCICS)	64K (for DFHMGT)
Each addt'l CICS	64K (for DFHMGT)

Ù Specify SPSIZE as small as possible, best: 0K

This shared space
- can only be used for shared partitions,
- should not be larger than the sum of all really required shared partition sizes.
- should always be specified as a multiple of 64K
- is the segment rounding space, if SPSIZE=0K specified

Ù Do not grossly oversize NPARTS specification

(maximum number of concurrently active partitions).

Costs 1 task and some bytes shared-24 per dynamic partition (even if unused)

Ù Check GETVIS-24 usage in shared space

e.g. via the "Storage Display" of the Interactive Interface.

Reduce the GETVIS-24 specification in the SVA command, if the high water mark after a real peak hour of operation is more than say 256 KB below the end of System-GETVIS (but only if somebody else may profit from it)

Ù Specify ample GETVIS-31 space in SVA command

Avoid that GETVIS LOC=ANY requests are satisfied from below.

Be generous. Selecting a high value does not harm, use e.g. a value of 6M or even more.
The SVA-31 space is required e.g. for all the pertinent VTAM buffers defined in your startup.

More Private Space (cont'd)

Ù Use the GETVIS command to check used GETVISs

GETVIS SVA, GETVIS Fx ...

31-bit GETVIS 'used' = "Any"-value minus "24-bit"-value

Do not risk any shortage in 31-bit space

Ù Adapt VTAM 4.2 buffers to your individual requirements, especially IOBUF in SVA-24

Issue a D NET,BFRUSE after a peak hour of operation and adjust the VTAM buffers accordingly to your specific environment. Long term, it's wise to avoid IOBUF extensions.

Refer to 'ACF/VTAM Resource Definition Reference' SC31-6498 for more information.

GETVIS-24 exploitation

„ Use the 'Storage Display' function of the IUI

to display the SYSTEM GETVIS Area occupations.

„ Always leave about 256K System GETVIS-24 free

(total space minus high water mark),

to be able to respond to dynamic requests

-> increase IPL SVA GETVIS-24 parameter if needed.

It is reasonable to especially observe VTAM IOBUF expansions (and avoid them later on). After that, SVA-24 usage should be more stable.

Mirror directories (which live until end-of-jobstep) may help to faster re-load a phase:

For FETCH/LOAD with DE (Directory Entry) mirror directories are only built for a jobstep, if still enough SVA-24 space is available (static partitions, for dynamic partitions Dynamic Space GETVIS is used).

VLA General Usage Aspects

General Aspects for VLAs in SVA-24/-31

Loading SVA-eligible phases into SVA-24/-31 areas, called VLA (Virtual Library Area)

Ù Be careful for loading phases into VLA-24

SVA-24 is a limited resource, in spite of all 31-bit applications and control blocks above the line (reduces the available private space below the line).

„ Never load all phases into SVA-24 which are eligible

„ Even be cautious to load all recommended phases of a product (IBM or vendors) into SVA-24

Ù Avoid SVA-31 eligible phases in SVA-24

„ Can be avoided by specifying big SVA PSIZE-31 value

„ Can be detected e.g. via console msg L169I

Ù Be generous for loading phases into SVA-31

SVA-31 is an ample resource, and only is a waste if modules are never used

„ Size of the VLA-31 is controlled via SVA PSIZE-31

Ù Check PSIZE-24 requirements in shared space via Librarian LD SDL

Reduce PSIZE-24 specification in SVA command, if too much space is unused in VLA (Leave at least 50K for service updates)

Ù Specify PSIZE-31 in SVA PSIZE=(____K,xxxxK)

If not enough VLA-31 space is available, phases are loaded into the VLA-24]

Any shipped value (meanwhile 3M) may be too low, if addt'l phases are loaded (e.g. vendor products)

Observe message
L169I 'Not all SVA-31 eligible phases loaded into SVA-31'

Ù Do an SVA-24 tailoring as discussed next

Performance Aspects of VLA

Ù Saves I/Os for loading of phases

- Put SVA eligible phases (relocatable and re-enterable) into VLA-24 or -31

or

- Put directory entries into SVA (via SET SDL, but without ',SVA')

But loading of phases can be enhanced:

- Proper LIBDEF chain setup (e.g. put a frequently used sublib into the temp chain, which is always searched first)

- Make sure that mirror directories can be built (LOAD/FETCH requests with DE= Directory Entry), used e.g. by CICS (System GETVIS-24 not exhausted)

- For frequent loads, use a 2nd-copy-library on Virtual Disk (created by the user at startup time)

- 'MDC'-caching the volume in VM (VM/VSE minidisk)

Ù Saves virtual (and real) storage

(if phase is used concurrently in several partitions).

This aspect becomes less and less important with bigger processor storage sizes.

Í Loading of phases into the SVA-24 does in general NOT give VSCR
(except the shared space would be required to exploit 1M rounding)

í These benefits have to be traded off with increased shared space requirements (usually a problem below the line)

SVA-24 Phase (VLA-24) Tailoring

VLA Background Information

Ù VLA load process

VLA loading can be done

- Via load lists (e.g. the $$A$SVA phase, which
  automatically loads phases during IPL)
  BUT: Tuning such load lists may be very dangerous, see below.

- Via SET SDL lists in a procedures
  and ',SVA'specification behind the phasename.
  Tuning SET SDL lists is not dangerous, see below.

Ù VLA re-load aspects

- Any phase loaded again (e.g. twice) occupies new space,
  an thus increases VLA space requirement

- There is no hint or warning that a phase is already
  in the VLA, upon re-load

- 'Old phase' remains in the SVA, since it may be already
  in use (with that load address) by a program

- LD SDL does not explicitly show this

- The directory entry itself is reused (SVA SDL= parameter)

í Do a LD SDL, before any private SET SDL procs are executed

Keep the list

í Only re-load a SVA-phase if changed/required

(e.g. during service)

í Before setting up SET SDL entries, check that a phase is not already in SVA

VLA-24 Tailoring

Optimize the loading of the Virtual Library Area (VLA) in the SVA-24

Ù Phase selection

General

- The fact that a phase is SVA eligible (reentrant) is not reason enough to load it into the SVA

- Monitor/determine those 'SET SDL phases' which in your system

  - never are required, OR

  - are only seldomly used AND need not reside for functional reasons in the SVA, and thus may be fetched from the VSE library.

If the usage of an SVA phase is not obvious from the module name, it is not easy to control its usage with SDAID, especially for phases loaded by the system at IPL time where SDAID cannot yet be active. SVA phases may be used by the VSE system just by branching into them via an internal table.
So, utmost care has to be taken when modifying IPL loaded phases.

Load especially those modules into the SVA which are concurrently used by several tasks/partitions.

Display

Use the LD SDL function of the Librarian to display SVA contents (SDL directory entries, SVA phases).

Ù Check specific CICS phases in the SVA-24

If CICS/VSE MRO is not used, the following phases are not required:
DFHCSEOT, DFHIRP, DFHSCTE (12192 byte)
If also CICS/VSE XRF is not used,
DFHCDDAN (8 byte)
you can fully bypass the $SVACICS load book in $0JCL.PROC

SVA-24 Phase (VLA-24) Tailoring ...

VLA-24 Tailoring (cont'd)

Ù More info

For more hints, refer to the following descriptions:

a) VSE/ESA, Guide to System Functions, SC33-6611,
'Loading Phases into the SVA'

b) The VSE Shared Virtual Area. By Daniel G.Arms.
Enterprise Systems Journal (ESJ) 03/91

c) VSE $SVA Tailoring. By Pete Clark, ESJ 01/91

This article (then for VSE/SP) shows how to carefully set up modified $SVA phases in order to change SVA loading during IPL.

In the essence, only the $SVAICCF phase is a potential, but only if ICCF would not be used at all:

$SVAphase	VSE/ESA Comment
$SVABAM	Since VSE/ESA 1.2, ISAM modules IJH..... are only loaded if ISAM is used
$SVAVSAM	VSAM load list (most SVA-24, some -31)
$SVAICCF	Required for ICCF, may be 'dummied' if ICCF not used at all (saves about 39K)
$SVASEC	Load list is only used if SEC=YES is specified, no need to compile a dummy one for SEC=NO
$SVACSC	NEVER TOUCH] Vital base supervisor SVC code]
$SVAVTAM	VTAM phases: 136 phases total, 7 (30K) in SVA-24
$SVACICS	CICS phases: 4 phases total, 4 (12K) in SVA-24
$SVAREXX	REXX phases: 12 phases total, 2 (12K) in SVA-24
$SVAASMA	High L.Assembler: 12 phases total, 2 (10K) in SVA-24

- The first set of $SVA phases is part of $$A$SVA, which is being executed automatically.
- The second set is explicitly called at VSE/ESA startup. Add other product specific $SVA lists only with care (often too big, and include 24 and 31-bit phases).

Gain 1M more Private Space

After having done first checks to reduce shared space below the line
...

Check chances for gaining 1 MB more private space

Be aware

Ù Total shared space in VSE must be always a multiple of the segment size (1M)

Ù All space VSE requires to round up total shared space below to an integer number of 1M segments is always reserved for shared partitions,
in addition to the SPSIZE specification, even if SPSIZE=0 has been explicitly specified.

Therefore, if your 'real' shared requirement would be an amount of DELTA above an integer number of MBs (e.g. 4.2 MB),

VSE then adds an amount of UPLIFT = 1 MB - DELTA (e.g. 0.8 MB).

Starting point

If you already have loaded privately any CICS or other phases into the SVA-24
(in order to exploit shared space required for the 1M rounding) ...

Start optimizing shared space WITHOUT loading them in VSE/ESA V2, since you may not require that in the same amount.

Gain 1 MB more (cont'd)

1. Determine by how many KBs you have exceeded a 1 MB boundary (DELTA)

a) General (more complex) case for UPLIFT calculation

This more complex case can be avoided by specifying SPSIZE=0K or as an exact multiple of 64K (see b))

Ù Calculate total space available for shared partitions (TSPS):

- Issue a MAP command after all shared partitions have started
- Add all sizes from lines for shared partitions (if any), plus the "UNUSED" space in the SVA-24 line:

```
SPACE        AREA              V-SIZE      GETVIS      UNUSED

  S            SUP               jjjK
  S            SVA-24            mmmK        lllK        nnnK
  S    shared  part-ID    xxxK          yyyK
       ...
  S    shared  part-ID    xxxK          yyyK
```

The Total Shared Partition Space (TSPS) or Actual SPSIZE results as

```
TSPS  =  all xxxK + all yyyK + nnnK
```

Ù Calculate the amount of shared space which was rounded up (UPLIFT):

```
UPLIFT = TSPS - (SPSIZE in IPL SYS, as a multiple of 64K)
```

Gain 1 MB more (cont'd)

b) Simple case for UPLIFT calculation

Requirement:
Specified SPSIZE = 0K
or
= the sum of all shared partition allocations
is a multiple of 64K]

Ù Issue a MAP command and take the value for

SPACE	AREA	V-SIZE	GETVIS	UNUSED
S	SVA-24	mmmK	lllK	nnnK

Ù nnnK is the amount of shared space which was rounded up (UPLIFT).

VSE/ESA 2.2 Enhancement: Msg 0J45I

'24/31-Bit System GETVIS Area has been rounded by xxK/yyyyK'

Actual rounding values are directly displayed.
Be aware that
- small roundings upward are always recommended
(avoid that any minor change requires a new segment 64K/1M)
- these rounding values can anyhow be only rough values
to apply to get 0 or small roundings

c) Calculate

DELTA = 1 MB - UPLIFT

The closer UPLIFT is to 1 MB, the better are the chances for gaining a full 1M segment in the next step #2.

2. Reduce, if you can, shared space by DELTA

Other shared space definitions/requirements (including shared partition sizes) should be tried to be reduced by DELTA:

You obtain 1 MB more private space.

Gain 1M more Private Space ...

If no reduction by DELTA is possible...

3. Check your SVA-24 (VLA) contents

From a total system point-of-view it may NOT beneficial to load all SVA-24-only eligible modules into the VLA-24. Often the mostly used ones provide nearly equivalent CPU- and Elapsed- time performance.

Restart optimization with Step 1.

- **You should be able to have 11M private space (i.e. 5M shared-24)**

  In some cases, even 4M shared-24 can be achieved.

If UPLIFT not needed for shared partitions...

4. Do the next 2 actions in the sequence as shown:

- **Move some eligible phases into the SVA-24,**

  instead of loading them into the virtual storage constrained partition below the 16M line.

  Especially for a CICS partition, the DSA and the partition GETVIS-24 area can be increased, by loading specific CICS modules into the SVA-24, as described in the CICS/VSE System Definition Guide, SC33-0706, Appendix A.

  Take a formatted CICS dump (CEMT Perform SNAP) as base for that, or better, refer to your NLT which shows all the permanently loaded modules and not also dynamically loaded ones. The default NLT modules are shown in 'CICS/VSE Resource Definition (Macro)', SC33-0709.

  If, for example, in a VSE/ESA 1.2 base installed system, you load only all the nucleus modules of the module list above, you require about 765 KB in the SVA-24 and obtain about 500K for the CICS DSA, provided this space was used to do a segment rounding.

  Do not add DFHMGTT to the SET SDL list, since this module is loaded into VIO.

  You also can put e.g. read-only assembler programs into the SVA.

- **Add this remaining space to System GETVIS-24**

  for general use by all partitions:
  Increase the IPL SYS GETVIS-24 specification.

Make Better Use of Private Space

- **Make sure, you have exploited that shared space, which is required to do a 1 MB segment rounding**

  Refer to the description under 3. on the last page.

- **Use VSAM Local Shared Resources (LSR) and carefully select subpool definitions.**

  Reduces I/O operations to VSAM files
  and saves virtual storage e.g. in CICS partitions.

- **Optimize VSAM LSR subpool definitions**

  Use the VSAM LSR statistics from e.g. the CICS shutdown.

- **Leave PGSIZE=2048 in CICS/VSE SIT**

  Saves virtual storage in the CICS partition
  by less fragmentation in the CICS DSA.

- **Use Autoinstall instead of TCT-defined CICS terminals**

  Saves virtual storage in the CICS partition
  when terminal not in use
  (640 byte + length of TCTUA (0 to 255 bytes))

VSE GETVIS Performance Aspects

Basics

- **GETVIS always rounds up any request to a multiple of an 'allocation unit'**

```
  16 byte in shared space
  16 byte in Dynamic space GETVIS Area
128 byte in private space
```

- **GETVIS fragmentation is caused by FREEVIS requests**

- **The performance of an individual GETVIS request depends on**

  - the current position of the GETVIS pointer (marks first possible free area)

  - the number of allocation units to be searched until a (contiguous) area is found

- **VSE allows up to 128 GETVIS subpools per partition**

  (none, 1.. 127, introduced long ago in VSE/AF 2.1.0)

  If no SPID is being given, the default subpool 'none' is used.

  - **This default pool may become very big**

- **Subpool properties**

  A subpool is initialized at the first GETVIS request for a subpool with SPID=name

  Areas for different subpools belong to different 4K pages

  Empty pages within a subpool are released

Basics (cont'd)

„ GETVIS search directions

GETVIS LOC=BELOW: bottom to top (low to high addresses)
GETVIS LOC=ANY : top to bottom

„ 'First Fit' is the implemented GETVIS algorithm

Means that the search is always started at the same end.
The first gap that can satisfy the request is being used
(a 'best fit' would be very expensive).

'Next fit' would start at the last position, but would result
in a higher working set, thus not used.

„ GETVIS is not able to determine 'orphaned storage'

- Orphaned storage results if no/no correct FREEVIS is done

- Orphaned storage is more harmful if in SVA
  (Partition areas are freed at end-of-jobstep)

- GETVIS has no information on usage and life of tasks and
  use of shared areas by other tasks

Only a VSE performance monitor (e.g. EXPLORE/VSE) can help.

í Worst case for GETVIS CPU-time consumption:

- all requests (means also many MBs) in same subpool
- single small free areas near the begin of a GETVIS search

Both situations cause long search chains

„ Partition GETVIS control info always resides at the end of a partition (31-bit eligible)

About 1 bit for 128 byte GETVIS, so 4K for 4M GETVIS.

Amount is calculated at VSE job startup

„ At GETVIS time (and at first reference of a page) no Page-in is required

(since 'no valid copy on PDS' is set for the page)

GETVIS Recommendations

Recommendations

- **Monitor your GETVIS areas, also to early detect 'GETVIS creep' (orphans)**

GETVIS partid ! SVA gives info on used/free storage with high water mark for BELOW and ANY

VSE/ESA 2.3 GETVIS enhancements:

GETVIS ...,ALL displays info on the allocated storage (below and above the 16M line) for the different subpools (in SVA or partition)
GETVIS ...,DETAIL in addition gives the addresses for the different subpools

- **Use and explicitly specify subpools, to separate small/big and short/long living GETVISs**

  - to reduce fragmentation
  - to shorten GETVIS searches (CPU-time)

  Naturally, this affects those programs directly issuing GETVISs.

- **Check applicability/possibility of using**
  - **GETVIS sizes as multiples of 4K**
  - **plus the PAGE=YES option**

  This results in
  - shorter GETVIS pathlengths, since space taken from the GETVIS FREE-chain of pages
  - faster FREEVIS, since clearing of VS is not required

  Before using PAGE=YES for 4K boundary alignments (mostly system programs), be aware that total GETVIS space requirements may increase

Recommendations (cont'd)

„ Avoid too frequent/unnecessary GETVISs

If it is easy and directly foreseeable, any program should try to GETVIS storage with less GETVIS calls.

This not only saves supervisor pathlength, but also saves space by less roundings to multiples of an allocation unit (16/128 byte)

„ Do correct FREEVISes, to avoid 'orphaned GETVIS'

„ GETVIS in CICS/VSE

Note that in CICS/VSE all space acquired via
GETMAIN FLENGTH>4K LOC=ANY

is using VSE GETVIS above the 16MB line,
since in CICS/VSE no DSA is available as in CICS/ESA for MVS.

Information APAR PN54891 describes this subject for CICS TS MAIN requests

„ Check use of 'Callable Cell Pool' services instead of doing individual GETVISs

- has same implementation in VSE as in MVS
- is a more specialized service
  - with shorter overall pathlength
  - which does not need non-parallel state

GETVIS Subpool Usage Examples

GETVIS subpools used by

- Supervisor
- VSE/POWER
- Librarian
- CICS
- user programs (if desired, is recommended)

Subpool name	Location	Purpose
IJBPRC	SVA-24	for static partitions Storing of JCL info by $IJBPROC (1 pool per partition via PIK)
	DS-GETVIS	for dyn. partitions
IJBPxx	_____	Storing of ___ info by _____
_____	PS-GETVIS24	Storing of POWER __-info
ICICSS	PS-GETVIS31	Storing CICS short-lived data (incl. TS MAIN],txn backout buffer)
ICICSL	PS-GETVIS31	Space for LE enclaves (w/o SIT XSTOR definition)
none	PS-GETVIS31	Storing CICS long-lived data (table mngr space)
I_____	_____	Storing of

PS = Private Space, DS = Dynamic Space

- Subpool names starting with 'I' are reserved for internal VSE (IBM) subpools

PART K.

App. B: VSE and SAP R/2 Performance

The following charts have been setup in cooperation with SAP AG, Walldorf, Germany.

VSE/ESA 1.3 Benefits for SAP R/2

Benefits for 24-bit R/2 applications

- **All VSAM buffers (LSR, NSR) above the 16M line**

  Most R/2 files are KSDS and use LSR

- **Multiple VSAM LSR pools for DIM exploitation**

  LSR statistics also directly from R/2 Tune-transaction

- **CICS 2.2 areas above the 16M line**
  High amount of DTB buffers required

  No spill-over to DASD, high partition size required
  More flexibility in VSE/ESA 1.4/2.1 (CICS/VSE 2.3) with
  DTB=AUX,FORCE

- **3990-3/6 extended caching functions**

  DASD Fast Write important for all updates

- **Larger VSAM physical blocksize (>8K, e.g. 3390)**

  RRDS roll file for planned 'swap-out' of session data,
  with 16K default VSAM CI-size

- **CICS Data Tables are beneficial for full-key KSDS requests only**

  R/2 system KSDSs mostly use partial key

Benefits for 31-bit R/2 applications

- **All benefits of 24-bit R/2 (see above)**

- **Nearly all R/2 applications reside above 16M**

  Only about 500 KB remain below, several MBs today

- **All R/2 internal buffers above the 16M line**

  Includes task work areas (at least 450 KB per 50 users)

- **R/2 roll buffer above the 16M line**

  Reduces the need for using the roll file,
  several MBs, up to 40M and more in MVS and now in VSE, max=128M

VSE/ESA 2.1 Benefits for SAP R/2

- **All benefits as for VSE/ESA 1.3 (see last foil)**
- **All system benefits from VSE/ESA 2.1**

  POWER performance improvements, VTAM 4.2 improvements, ...
- **Major VSAM file control blocks above the 16M line**
- **Possibility of DTB=AUX,FORCE for CICS/VSE 2.3**

  No need then to define a huge SAP CICS partition
- **H/W Data Compression for all SAP 'user KSDSs', incl. ABEZ ('Beleg Datei') KSDSs for R/2 6.0**

  Reduces VSAM 4 GB file limitation.

  ESDSs only applicable if only updates at end-of-file are done
- **Turbo Dispatcher support of direct benefit only if**

  several separate SAP CICS partitions are setup (rare)

  a data base is used (in a separate VSE partition)

VSE/ESA and SAP R/2 Migration

Statements by SAP AG

- **R/2 Rel 5.0 CPU-time increases vs Rel 4.x**
  - **A function of application or transaction type(s)**
  - **Individual customer assessments by SAP Basis Consulting**

Refer e.g. to R/2 5.0 customer letter by SAP AG (next foil)

- **R/2 Rel 6.0 CPU-time vs Rel 5.0:**

  No measurable delta expected for same setup

IBM Recommendation

- **Separate the CPU-time impact by R/2 Rel 5.0 from VSE/ESA impact**

- **Do not change VSE and SAP release at same time.**

  Change SAP release either before or after VSE migration

R/2 5.0 Customer Letter by SAP AG

Resource Consumption with SAP R/2 5.0

The following is a direct translation of an SAP customer letter, sent by SAP AG to VSE SAP R/2 customers:

'Now, where a bigger number of customers is in production with SAP R/2 5.0, we want to tell you our experiences regarding the increased resource requirements.

Due to the different setup and usage of the SAP systems, naturally, a bigger variation results. Nearly all customers use new components or use existing functions more often. A direct comparison between release 4.3 and 5.0 is therefore possible only in a limited fashion.

In order to demonstrate the scope of variation, sample results of 3 customers are shown:

	R/2 Applications	CPU-time	Real St.	DASD St.	
Case 1	RF, RA, RK-S	0%	80%	30%	
Case 2	RF, RA, RK, RM-MAT	5%	--	33%	
Case 3	RF, RA, RM-MAT, PPS, RV	30%	80%	--	

In the average, the following holds:

- **CPU-time increase for Online is between 20% and 40%**

- **Required real storage (working set) increase is between 80% and 100%**

  (It has to be defined how much of this additional real storage was really required for SAP programs/areas and how much of it were used for better performance via Data In Memory.'WK')

- **DASD storage increase is between 15% and 20%**

  We offer to determine the individual increase of your installation during the migration, as part of our Base Consulting activities.'

CICS Partition Size for SAP R/2

DTB Situation

- **DTB buffers are allocated above the line, if 31-bit GETVIS is available**

  in spite of DTB=AUX

  (Both for 24 and 31-bit appl's, holds for R/2 4.x, 5.0 and 6.0)
  No spillover to TS AUX (DASD). Same as in CICS/ESA for MVS/ESA.

- **I/Os saved at cost of DTB buffer requirements**

  Increased CICS partition size, VSIZE, Page Data Set.

- **SAP background transactions may run a long time, but syncpoints may not be possible**

  Much data in a single LUW, running for several minutes:
  e.g. if many not further specified items are paid in a single sum

- **High DTB buffer requirements possible**

- **Here comes the problem:**

  If GETVIS above the line is exhausted, CICS issues DFH0505 'Short on Extended Storage' and more or less stops processing

  DFH0506 'CICS is under stress' also appears, but purging of transactions/areas below the line does not help here.

  (Only users with a CICS GETVIS monitor may proceed, but only after cancel of the task requiring the excessive DTB buffer space.)

- **VSE/ESA 1.4/2.1 with CICS/VSE 2.3 allows DTB=AUX,FORCE**

  When DTB buffer (DBUFSIZE) full, records are spilled to disk (done asynchronously), 16K CI-size recommended
  CAUTION: PTF for APAR PN79087 required: UN86764/UN86765

CICS Partition Size for SAP R/2 ...

What to do with DTB?

In case DTB=AUX,FORCE of CICS/VSE 2.3 is not used:

- **Start with a big CICS partition for R/2 (say 80M)**

  This is the same value as is and was recommended for MVS/ESA

  As a rough indication, you do not need more storage than you had as DFHTEMP on DASD in the pre-VSE/ESA 1.3 environment.

  Actually, even 120 MB may turn out to be too small

- **Monitor 31-bit GETVIS high water mark carefully with 'GETVIS Fx'**

  over a longer period of time, including month-end or year-end work.
  Make sure you catch the high activity DTB peak, which only may exist for few minutes.

- **Reduce CICS partition size later (if you want)**

  - **It is not reasonable to only slightly pass the 16MB line**

- **Real storage aspects**

  These DTB data are written into fresh pages and later on are no more referenced, except in case of a backout.

  Therefore, only asynchronous page-outs occur, which should not cause a CICS paging problem.

  Due to this specific use of virtual storage...

  - **It is not required to back up these additional 'DTB megabytes' with real storage**

  - **More real storage is required/beneficial for bigger programs, and for (more) Data in Memory**

SAP R/2 6.0 for VSE (Summary)

R/2 6.0 Enhancements for VSE

- **Release 6.0A**
  - **Enhancements in base component, function-wise equivalent to R/2 5.0F**
  - ADD-installation possible from 5.0F
  - 31-bit only
- **Enhancements aimed at**
  - **higher maximum R/2 throughput**
  - **longer Online availability**
    - thru smaller required batch window by more parallel batch

    (but no Batch updates concurrent to Online updates)
- **Enhancements (Summary)**
  - **APLZ-switch**
  - **ABEZ-split**
  - **Multiple parallel update tasks**
  - **Several BTCIs**

R/2 6.0 VSE Performance Enhancements

APLZ-switch (update queue file, 'Protokoll-Datei')

(VSAM ESDS for all R/2 releases)

„ **Switching on the fly now possible**

e.g. automatically at 12 pm

Í Online can continue with newly opened APLZ

Í SAPLOGU log utility can backup/reorg closed APLZ-file
or SAPREPU can create reports in batch

ABEZ-split (document file, 'Beleg-Datei')

A VSE specific solution w/o requiring SAP/MRO component for MVS

„ **VSAM ESDS in R/2 5.0 or 6.0 as 'single ABEZ',**
replaced by
VSAM KSDSs in 6.0 as 'multiple ABEZ'

(BELx (+BIBx) pairs, 1 per doc-type, up to 12)

„ **Makes the VSAM 4 GB file limit less restrictive**

Change from ESDS to KSDS allows ...

„ **VSE/ESA 2.1 H/W compression for 'multiple ABEZ'**

„ Sequential inserts into KSDS gives high CI split activity

Usage of (small) keyranges recommended

„ **ABEZ reorganization can be done independently**

„ Database recommended by SAP

R/2 6.0 VSE Performance Enhancements ...

Multiple parallel update tasks ('Mehrfachverbucher')

All update tasks must run in 'V1SA-CICS' (MVS)

„ **Exploits multiple update queue files (ABEZ)**

„ **New table: Multiple Update Classes (MUPC)**

- sequential standard updates (V1SA, 1 task), as before
- parallel updates for SAP transactions (VXSA, n tasks)
- updates for specific SAP transactions (VLSA, 1 task), not eligible for VXSA

MUPC must be defined installation dependent

Several BTCIs (Batch Input files)

Multiple input files from batch allow ...

„ **Overlap processing of 'mass data' for D1SA tasks:**

- batch input to a BTCI
- processing of another BTCI by D1SA dialog tasks

R/2 6.0 Enhancements...

í **Of specific value to high transaction volumes**

Potential Addt'l Tuning Hints

Some ideas

If you want to max out your SAP R/2 performance,
on top of the shipped R/2 layout and setup parameters,
do the following:

Ù Use SAP R/2 6.0 if you have a 'big' SAP production
and want to benefit from new performance/capacity features

Ù Use the usual VSE system and CICS performance monitors

and use R/2 specific tools on top (TUNE transaction)

Ù Tune your system as non-R/2 customers would do

Ù Use a VSE Virtual Disk for Journaling

In case of a CICS crash, SAP R/2 is doing a cold start.
Journaling is 'only' used for Dynamic Transaction Backout.

Ù Check the following R/2 specific items

„ Use VSAM LSR for all files that are not BROWSEd and concurrently updated

„ For VSAM LSR, in any case, use multiple LSR pools

„ Use reasonably high BUFND, BUFNI values for R/2 Batch jobs

„ Check whether any file can be defined on a Virtual Disk

E.g the VSAM work file LIST

„ Check the CI-sizes for the VSAM files

„ Number of work areas

„ .....

--- Foil under construction ---

PART L.

App. C: VSE/ESA 2.3 Perf. Enhancements

VSE/ESA 2.3 Performance Enhancements

„ All Performance PTFs shipped after V2.2 GA

- GETVIS display enh.
- POWER PTFs (LTA...)
- LE PTFs incl. CICS 2.3 LESTG

„ Turbo Dispatcher Enhancements

New supervisor services for vendors
STOPQ (Quiesce) CPU
QUERY TD Display

„ VSAM Enhancements

VSAM >4 GB KSDS files
Multiple Catalog Backup
Slightly reduced compression pathlength

Both for H/W assisted compression, and without

„ POWER Enhancements

Moved to VSE/POWER part

Bigger DBLK default sizes
Bigger PNET default buffer size
POFFLOAD PICKUP

Cont'd

VSE/ESA 2.3 Performance Enhancements (cont'd)

- **Misc. Enhancements in Base**

  Startup message on 64K rounding

  Improved sequential READ performance for
  - **FASTCOPY (OPT>1)**
  - **LIBR**

  Up to 32 sublib entries in LIBDEF statement

  Always performance optimal for library search:

  the TEMP LIBDEF contains ONLY the sublibs needed, including
  IJSYSRS.SYSLIB and others from the PERM LIBDEF

  Shipped GETVIS-31 increased from 3M to 6M

- **Enhancements for/in Components**

  LE/VSE 1.4 in base

  TCP/IP for VSE/ESA in base

  Refer to
  'VSE/ESA TCP/IP Performance Considerations'

  DB2 Server for VSE 5.1.0 (new SQL/DS level)

  DB2 VSAM Transparency for VSE/ESA

- **IXFP/Snapshot for VSE/ESA**

  Discussed in 'VSE/ESA I/O Subsystem Performance Considerations'

- **More Info**

  - IBM VSE/ESA Enhancements, Version 2.3, SC33-6629-01

VSE/ESA 2.3 TD Enhancements

„ New Supervisor Services for Vendors

With the new TD level, additional performance optimized services for vendors have been provided. They help in order to save non-parallel CPU-time and thus to reduce the Non-Parallel Share NPS.

„ Quiesce CPU

Problem

Dependent on the workload it may be necessary or beneficial to stop a processor (CPU) to avoid the overhead of an additional CPU that can't be exploited or is not required.

However, VM/ESA V=R guest environments with any 'not started (stopped)' CPU will have no I/O assist for dedicated devices. So the VM overhead may increase and not allow to benefit from a stopped processor.

Solution

A CPU can be quiesced via a new command 'STOPQ'.

Such a CPU will no longer participate in processing the workload. The overhead of the CPU, that is not required, can be avoided, and the VM/ESA guest continues to run with I/O assist.

Performance Results

Refer to next foil

VSE/ESA 2.3 TD Enhancements ...

QUIESCEing a Processor under VM

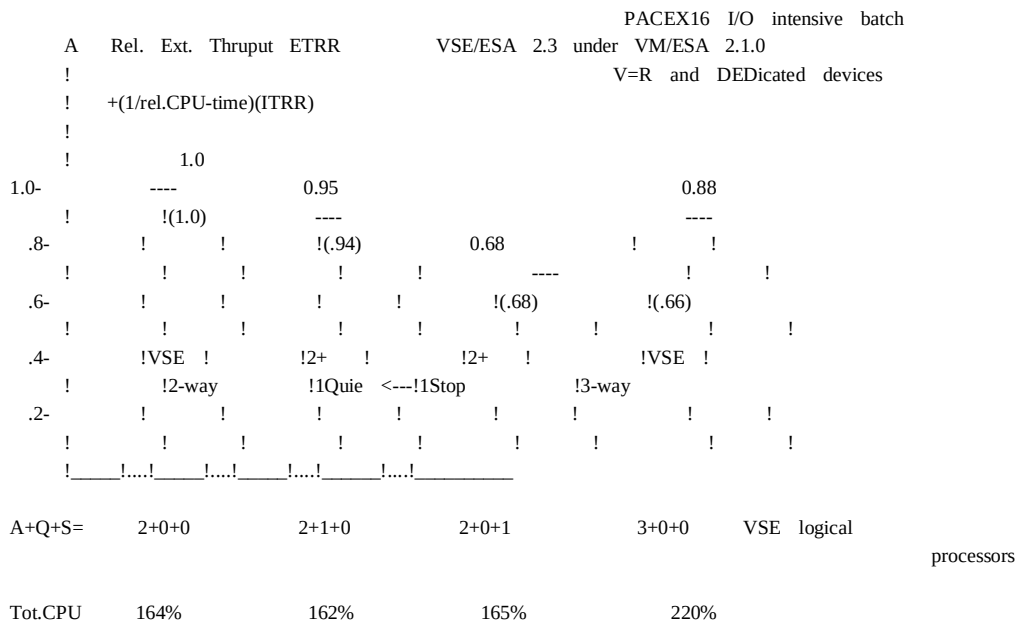
Ù Background

- A STOPped (INACTIVE) processor of a preferred VM guest (V=R/F) causes total loss of VM IO assists

  -> potentially significant increase of total CPU-time in case of DEDicated devices

Ù VSE/ESA 2.3 TD enhancement: QUIESCE = STOPQ

TD allows to STOP a processor w/o losing IO assist
(required native CPU-utilization on QUIESCEd processor: <<1%)



Í To QUIESCE (STOPQ) a processor instead of STOP gives higher throughput AND lower CPU-time

VSE TD guest 'stays in IO assist'

Delta and benefit is smaller
- if workload less I/O intensive
- if not all DASDs DEDicated

VSE/ESA 2.3 TD Enhancements (cont'd)

„ QUERY TD Enhancements

The QUERY TD command provides additional information concerning the workload:

Spin Share:

(SPIN_TIME) / (SPIN_TIME + TOTAL_TIME)

This is the share of time spent by processors in so-called spin-loops.

Overall utilization sum:

(TOTAL_TIME + SPIN_TIME) / ELAPSED_TIME

This value corresponds to the sum of all individual processor utilizations, which can add up to n x 100% (native)

NP Utilization:

(NONPARALLEL_TIME / ELAPSED_TIME)

This value is additional info to the well known 'Non-Parallel Share' NPS (or NP/TOT). It is the utilization of the non-parallel status and can reach at most 100% (native).

It is a good indicator of the remaining potential for achieving more total throughput, especially with more processors

VSAM KSDS >4 GB Enhancement

„ New Parameter for DEFINE CLUSTER:

EXTRALARGEDATASET (or XXL)

The salient internal change is that the former 4-byte key field now is treated as 4-byte relative CI-number.

„ Available for most KSDS uses

For Keyed access

- Not for RBA access
- Not for CNV access

Not for KSDSs defined with

- IMBED
- KEYRANGE
- REUSE

or - UNIQUE

„ Theoretical new maximum size of 492 GB

Still 4GB per volume (i.e. 64K tracks x 64KB).

With 123 volumes (extents) you get 123 x 4 GB = 492 GB

VSAM KSDS >4 GB Enhancement ...

VSAM >4 GB KSDS Performance Aspects

„ **No measurable pathlength increase for VSAM KSDS accesses for total production loads**

Applies both to non-XXL and to XXL KSDSs

„ **Distribution of an XXL file across logical volumes via KEYRANGEs is function-wise not possible**

Also applies to OS/390 (MVS).
Refer also to the following topic

Balancing of Multi-Volume Files across Volumes

„ **Balancing of a single huge/heavily used VSAM file**

- is important for non-simulated real disks and simulated disks for Internal Disk
- is not so much required for simulated volumes in RAID-5/6 I/O subsystems

„ **Balancing via KEYRANGE is often not easy**

In practice, reasonable keyranges are sometimes hard to specify

„ **Use of allocated VSAM space for multi-volume files (w/o key-ranges)**

- Uses first device first
  - primary allocation
  - secondary allocation(s) (if specified)
- Uses next device
- ...

í **Multi-volume files are not easy to be reasonably balanced across logical volumes**

VSAM Multi-Volume Files

General Recommendations for Multi-Volume Files

í For any access method (incl. VSAM) ... it is better to distribute data on more volumes

(select e.g. smaller logical volumes if simulated)

Reason is
- potentially less hot 'hot spots' (better balancing)
- potentially reduced msec/IO by smaller IOSQ time

Applies to all types of VSAM files, not only multi-volume

í If you want to split reasonably non-KEYRANGE KSDSs across multiple volumes ...

- Use a separate VSAM DASD space class for the file(s) where balancing is desired/required

- Specify a primary extentsize which essentially is equal to (maximum file size/number of volumes)

- Do not specify a secondary allocation (since that would always mean filling up of a VSAM extent)

More VSAM Enhancements

Ù Multiple Catalog Backup

Backup VSAM files of multiple BACKUPs and catalogs on 1 tape via the new option 'NOREWIND'.

This is a functional enhancement, which also assists to better exploit 3590 tape cartridge capacities.

Ù Slightly reduced Compression pathlength

With the new VSAM release 6.3.0 the VSAM pathlength for data compression and expansion was slightly reduced.

COMPRESS and EXPAND requests

This applies both to the case where compression is H/W assisted and also to (mostly non-IBM) processors w/o this assist.

Startup Message on 64K Rounding

Background Info on VS Management

Rounding (up) occurs for 1K, 4K, 64K, and 1M reasons

as extensively described here in 'Appendix A: Space Optimization'

Regarding the space BELOW the 16M line:

- 64K roundings (1K to 63K) is added to System Getvis-24

- 1M roundings (rounded themselves to multiples of 64K) are added to SPSIZE (even when set to 0)

New IPL message facilitates VS tuning

0J45I 24-BIT SYSTEM GETVIS AREA HAS BEEN ROUNDED BY xxK
31-BIT SYSTEM GETVIS AREA HAS BEEN ROUNDED BY yyyyK

xx is in the range of 1 to 63K,
yyyy is in the range of 1 to 1023K.

xxK is the (approximate) value which you would need in order to save a 64K upward rounding in the shared space below the line.

Do a VS tuning, as described in Appendix A.

Improved FCOPY and LIBR Performance

Improved FCOPY and LIBR seq. READ Performance

Ù Background

For some utilities under certain conditions, so far the SEQuential indication for cache handling was not set in all corresponding ECKD channel programs.

For VSE/ESA 2.3 the SEQuential indications are set in all cases, in order to benefit from the pre-staging of the I/O subsystems for DASD READs.

Ù FCOPY DUMP VOLume Results

Sequential improvements apply to all OPTIMIZE >1 specifications for DUMPing an ECKD attached volume. So far, for OPTIMIZE=1 (default) SEQ was used.

In the following case,

```
DUMP VOLUME OPTIMIZE=2 IV=VSAM01 NPOPROMPT NOVERIFY
```

dumped 3358 3390-tracks from an Internal Disk volume to a 3490 tape.

	VSE/ESA 2.2	VSE/ESA 2.3	Delta (SEQ)	
Elapsed Time (Rel thruput)	50.2 sec (1.00)	39.6 sec (1.27)	-21% +27%	
#DASD I/O msec/IO	1805 10.82 msec	1805 0.77 msec	 14x	

- 14 times faster READ I/Os from DASD resulted in 27% higher throughput
- About half of the total time spent in DASD READ was saved in the Elapsed Time of the total job
- Runs were done on a 2003-116 using Internal Disk
- Improvement for OPTIMIZE>2 may be lower.
  No change for OPTIMIZE=1

í 27% faster BACKUP of a DASD volume

Improved FCOPY and LIBR Performance ...

Ù LIBR BACKUP Results

Sequential improvements apply to all LIBR Read functions for ECKD attached disks.

In the following case,

BACKUP S=IJSYSRS.SYSLIB T=990 RESTORE=ONLINE

was used to backup a sublibrary from a cached 9345 volume to a 3480 tape unit:

	VSE/ESA 2.2	VSE/ESA 2.3 (SEQ)	Delta	
Elapsed Time (Rel thruput)	56.0 sec (1.00)	36.8 sec (1.52)	-33% +52%	
#DASD I/O msec/IO	3409 10.97 msec	3409 5.57 msec	 2x	

- 2x faster READ I/Os from DASD gave 52% higher thruput (includes non-sequential LIBR I/Os)
- Runs were done on a 9672-R11 with cached 9345s
- Scattered VSE sublibs may benefit less

Í 52% faster BACKUP of a VSE sublibrary

VSE/ESA and Number of I/O Devices

Background Info

Ù VSE/ESA allows up to 1024 ADDed devices

Local SNA terminals NOT to be ADDed in IPL procedure.
Local BTAM devices must.

The number of supported devices is determined by the IODEV operand of the IOTAB Supervisor Generation macro (Refer to the VSE/ESA Planning manual).

Ù BTAM-ES so far did not support IODEV>254

A supervisor generated with IODEV>254 did NOT support BTAM-ES.

Some VSE sites use a BTAM vendor product that allows IODEV>254.

E.g. BTAM-ESA by DPE & Associates

Note that BTAM is no more supported by CICS Transaction Server (VSE/ESA 2.4)

Ù BTAM-ES now (09/98) supports IODEV>254

APAR DY44756 (PTFs UD50634/UD50635) causes BTAM to use a VSE macro to access VSE control blocks.

Consider also PTF UD50750, replacing UD50635 (APAR DY44850).

Note that these PTFs do NOT make it possible to define >254 devices to a BTAM application.

Ù VSE/ESA V2 always comes with 2 supervisors

Both are 'MODE=ESA':

$$A$SUPX	IODEV=254	'old control block system'
$$A$SUPI	IODEV=1024	'new control block system'

For Base Install and most production images, SUPX is used.

VSE/ESA 2.3 Startup Variations

Observed Shared Sizes below the 16M line

Standard Boeblingen performance testing system (9672 CMOS),
not shared-space-optimized for regression reasons.

Supervisor Dispatcher Variation Case	SUPX SD	SUPX TD a)	SUPI TD b)	SUPI TD c)	TD +768 ADDs d)	
Resulting CHANQ	255	255	378	4986		
Actual SPSIZE x (= UNUSED, here)	64K	=	=	448K*)		
VPOOL		64K	=	=	=	
SLA		+108K	=	=	=	
Tot xy		=172K	=	=	=	
GETVIS USED y	1040K	1044K	1048K	1236K		
Tot x	=1660K	1632K	1616K	1788K		
PGM AREA USED y	1352K	=	=	=		
Tot x	1544K	=	=	=		
SDL list xy	32K	=	=	=		
MAP SUP V-SIZE xy	624K	652K	668K	1136K		
Shared-24 (sum x)	4096K	4096K	4096K	5120K*)		
Total used (sum y)	3220K	3252K	3272K	3928K		
Delta		-32K	Base	+20K	+676K	

- Normal startup includes 177 devices ADDed.
  For case d), 768 DASDs were ADDed on top
- SYS CHANQ=n not specified, system calculated values
- Sizes from IUI '363-panels' and MAP
- Actual SPSIZE here is UNUSED in MAP, since no shared partitions allocated
- SLA is the 'Label Work Area' in virtual storage

*) Here, no efforts done to compensate the 1M rounding via SVA (SET SDL)

- Areas illustrated in same sequence in separate chart

- **TD required 32K more than SD**

  (b) vs a), 4K in GETVIS USED, 28K in SUP area)

- **'More devices' requirements, see next foil**

  (c) vs b) and d) vs c))

Space Requirements for More Devices

Shared Space Requirements for More Devices

- ### Requirements for CHANQ

  - #### 32 byte required per CHANQ entry

    Note that a bigger channel queue is only required if more I/Os are active at any instant of time.

- ### IODEV requirements (1024 vs 254)

  Total SUP area with SUPI (at same IPL procedure) is bigger than SUPX:

	SUPX	SUPI	Delta
Base supervisor (from LIBR display) (380K) (390K) (+10K) (as on disk, just for info)			
Total SUP area (MAP: SUP V-SIZE) 652K 668K +16K			
- The 16K here also include a CHANQ value increased by 123. With 32 byte/entry, this means about 4K.			

  - #### About 12K more required for SUPI

- ### Requirements for ADDing devices

  ADDing more devices, will need some more shared space-24:

Area (768 ADDs)	Delta	Approx. Delta/ADD
SUPVR area (SUP V-SIZE 324K -144K CHANQ)	420 byte (PUBs)	
SVA-24 GETVIS USED 188K	240 byte	(PUB-extensions)
Total	512K	660 byte

  - #### About 660 byte required per ADDed device

Recommendations for More Devices

- **Try to use VTAM instead of BTAM terminals**

- **Use SUPX, as long as 254 devices sufficient**

  Needs 12K less space, at same CHANQ value

- **Switch over to SUPI, if required**

  IPL with LOADP ..P and say STOP=SUP,
  in order to be able to overwrite the supervisor name.

  No need to generate an own supervisor

- **Monitor CHANQ HIGH-MARK vs MAX via SIR**

  You may reduce CHANQ via SYS CHANQ if MAX too ample

- **Re-use/re-allocate any increased UNUSED space**

  It may be a good idea to let 64K unused (if possible),
  to not undergo a 64K rounding at minor startup changes

- **Do a VSE Space Optimization, as shown in Appendix A**

PART M.

App.D: Parts moved, or in other documents

Parts moved, or in other documents

Ù VSE/ESA 2.4 Perf. Enhancements

This subject always was contained as Appendix in

'IBM VSE/ESA CICS TS Performance Considerations'

Ù VSE/ESA Sequential Disk Files

Was moved into the new VSE/ESA 2.5 document

Ù VSE/ESA Librarian

Was moved into the new VSE/ESA 2.5 document

EOD/HAND

EOD End of Document

HAND Have a nice day