DFSORT/VSE - The Complete Sorting and Reporting Solution

DFSORT/VSE 3.4

Holly Yamamoto-Smith and John Burt

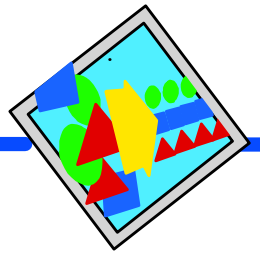
IBM Storage Systems Division
5600 Cottle Road
San Jose, California 95193

email: hollyyam@us.ibm.com
email: burtjm@us.ibm.com

www.ibm.com/storage/dfsortvse

RETURN TO INDEX

Reminder



DFSORT/VSE 3.4

References in this presentation to IBM products, programs, or services do not imply that IBM intends to make these available in all countries in which IBM operates.

Any reference to an IBM or other product, program or service is not intended to state or imply that only that product, program or service may be used. Any functionally equivalent product, program or service that does not infringe any of IBM's intellectual property rights, or other legally protected rights, may be used instead of the IBM product, program or service. Evaluation and verification of operation in conjunction with other products, except those expressly designated by IBM, is the user's responsibility

IBM may have patents or pending patent applications covering subject matter in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to the IBM Director of Commercial Relations, IBM Corporation, Purchase, NY 10577, USA.

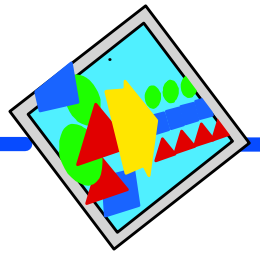
The information contained in this document has not been submitted to any formal IBM test and is distributed on an 'AS IS' basis without any warranty either expressed or implied, including, but not limited to, the implied warranties of merchantability or fitness for a particular purpose. This disclaimer does not apply if inconsistent with local law. The use of this information or the implementation of any of these techniques is a customer responsibility and depends on the customers ability to evaluate and integrate them into the customer's operational environment. While each item may have been reviewed by IBM for accuracy in a specific situation, there is no guarantee that the same or similar results will be obtained elsewhere. Customers attempting to adapt these techniques to their own environments do so at their own risk.

Any performance data contained in this document was obtained in a controlled environment based on the use of specific data and is presented only to illustrate techniques and procedures. The results that may be obtained in other operating environments may vary significantly. Users of this document should verify the applicable data in their specific environment.

Some of the information in this presentation describes possible future enhancements to this product. IBM may choose to modify, withdraw or not make available these future enhancements. In receiving this information, you should not make any changes in your business operations on the basis of information disclosed or discussed by IBM. If you make any changes, it is understood that these changes are made soley at your own risk, cost and expense.

For more information on the test environment and results, have your IBM field representative contact the DFSORT Hotline electronically at sort@vnet.ibm.com.

Test Environments



DFSORT/VSE 3.4

Performance measurement test runs for DFSORT/VSE Version 3 Release 4, DFSORT/VSE Version 3 Release 3, and Sort/Merge V2R5 were completed using the same test environments. The test environments included:

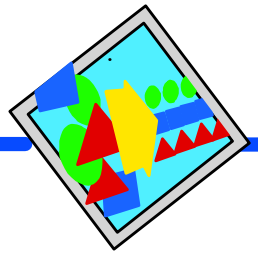
- Size of data space area: from 5 MB to 26 MB
- Size of GETVIS area: from 5 MB to 26 MB
- Size of partion program area: from 64 KB to 5376 KB
- FLR and VLR files: from 3.1 MB to 67 MB
- Number of records: from 1000 to 220000
- Block size: from 6400 to 56664 bytes
- Record length for FLR: 320 bytes
- Average record length for VLR: 320 bytes
- Keys were in random order and were 8 bytes in length
- For both DFSORT/VSE releases and Sort/Merge V2R5, the SVA eligible modules were loaded in the SVA
- VSE/ESA 2.2.1 operating system was used

All measurements were run on an IBM 9672 Model R73 with 512 MB of central storage, in a stand-alone VSE/ESA environment. All of the input, output, and work files resided on IBM 3390-2 disk storage connected to an IBM 3990-3 storage control unit. Both volatile cache of 256 MB and non-volitile disk storage Fast Write cache of 4 MB of the 3990 Model 3 disk storage controller were activated.

The actual performance characteristics that may be experienced by any specific user or for any specific file depends on many factors, including record length, file size, and DFSORT/VSE and Sort/Merge V2R5 storage options; noticeably, in a multitasking environment, elapsed time results are application-profile and workload dependent. So, the results may differ from user to user.

IBM does not represent nor warrant that users will experience the same changes in performance characteristics observed in these examples.

Trademarks



DFSORT/VSE 3.4

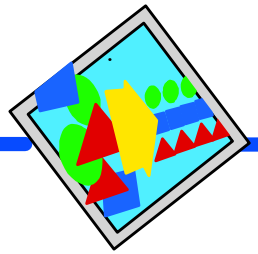
The following names, which are used throughout this presentation, are trademarks or registered trademarks of the International Business Machines Corporation.

- **DFSORT**
- **VSE/ESA**
- **IBM**
- **Language Environment**

Non-IBM trademarks used in this presentation:

- **IEEE and POSIX are trademarks of the Institute of Electrical and Electronics Engineers, Inc.**
- **X/Open is a trademark of X/Open Company, Ltd.**

Abstract



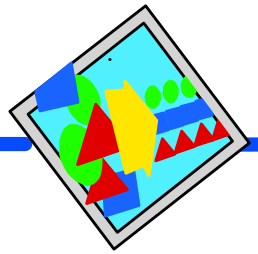
DFSORT/VSE 3.4

Find out what we've done with our latest release of DFSORT/VSE! With every release, we get faster and faster and add productivity enhancements to make your job easier! DFSORT/VSE has:

- **FASTER Sorting, Copying and Merging**
- **EASY ICETOOL reporting**
- **FLEXIBLE data handling and output formatting**
- **NEW Year 2000 features**

DFSORT/VSE makes good sense for you and your business! Come and find out why!

DFSORT/VSE Evolution



DFSORT/VSE 3.4

Increased Value for You

DFSORT/VSE 3.1 - 9/94

- 31-bit addressing
- Getvis sorting (FLR & VLR)
- Dataspace sorting (FLR)
- Secondary allocation for VSAM Managed Workfiles
- STXIT support

DFSORT/VSE 3.2 - 10/95

- Improved workfile use (multivolume SAM ESDS)
- Improved sorting algorithms
  - GETVIS sorting
    - Incore and Non-incore
    - FLR and VLR
  - Dataspace sorting
    - Non-incore
    - FLR
- ICETOOL

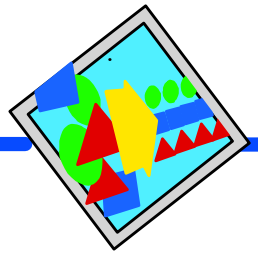
DFSORT/VSE 3.3 - 2/97

- Improved sorting algorithms
  - Getvis sorting
    - Incore, VLR
  - Dataspace sorting
    - VLR (new!)
    - Incore, FLR
  - COPY
  - MERGE
  - Tape
- File Management Support
- Year 2000 features
- National language support

DFSORT/VSE 3.4 - 5/98

- Improved performance
  - Getvis sorting
  - Dataspace sorting
  - Partition sorting
  - COPY
  - MERGE
- Additional Year 2000 formats
- Online message explanations (OME)
- OUTREC enhancements
- INCLUDE/OMIT enhancements
- ZDPRINT feature
- STXIT Improvements
- Full Year 2000 date formats
- OPTION PRINT=CRITICAL
- ENDEOD=YES|NO

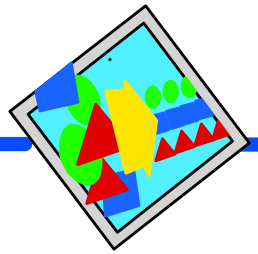
Performance Enhancements with DFSORT/VSE 3.4



DFSORT/VSE 3.4

- Dataspace Sorting
- Getvis Sorting
- Partition Sorting
- Copy and Merge Enhancements

Summary of DFSORT/VSE R4 Performance Enhancements



DFSORT/VSE 3.4

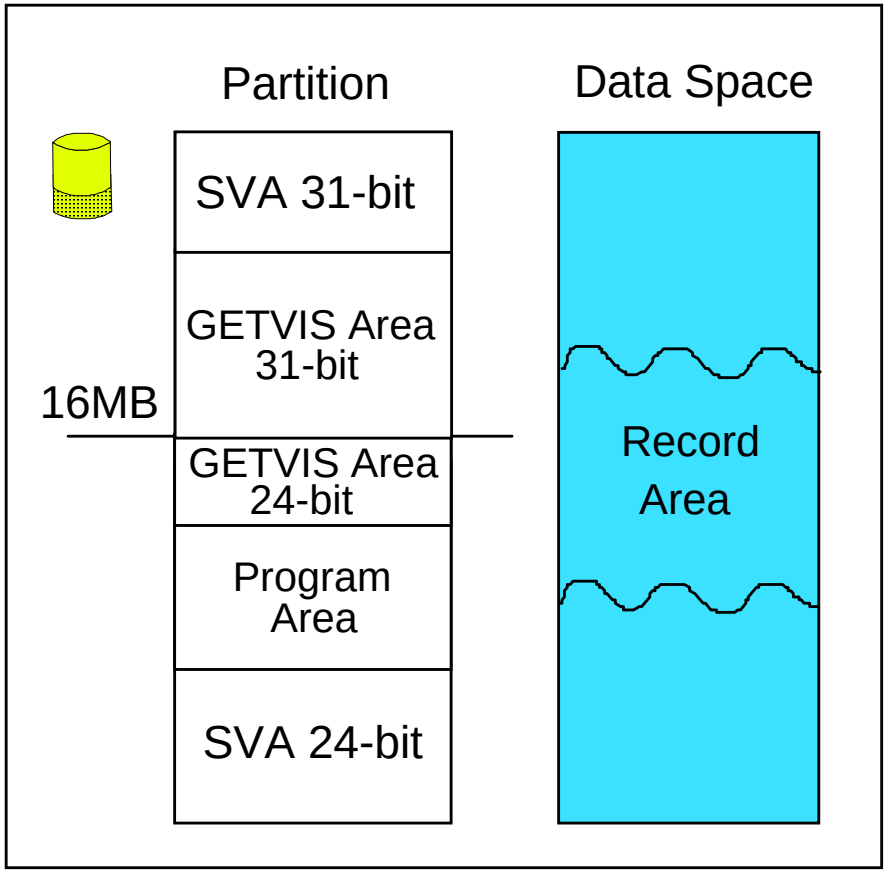
- **Enhanced data processing methods for:**
  - Dataspace sorting (incore and non-incore)
  - Getvis sorting (incore and non-incore)
  - Partition Program Area sorting (non-incore)
  - Copy and Merge applications
- **Enhanced input/output processing for:**
  - Non-VSAM input and output files
  - Variable-length record SAM output files
- **ECKD support for input, output and work files**

Dataspace Sorting

DFSORT/VSE 3.4

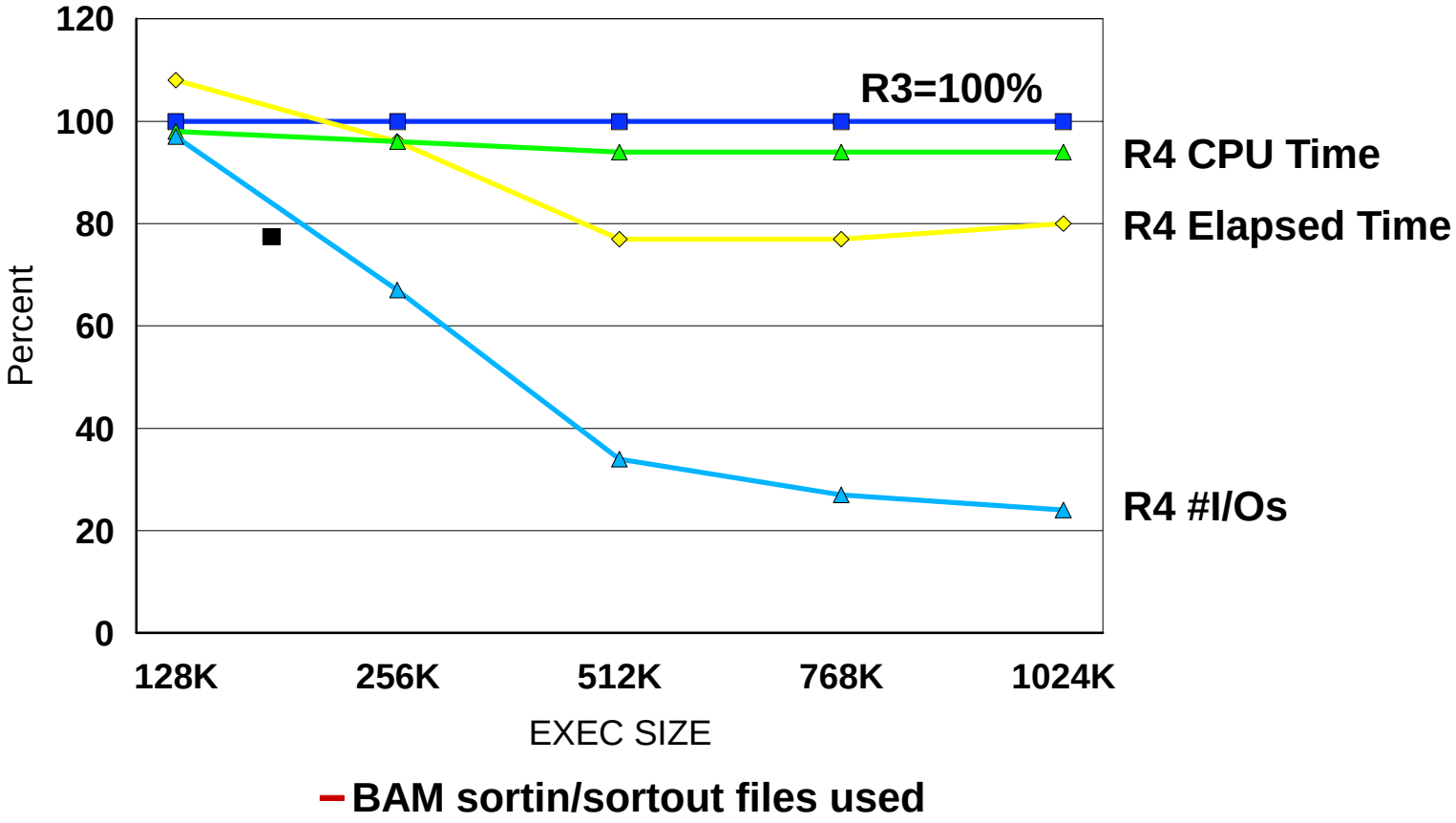
Dataspace sorting is used if DSPSIZE=n

Dataspace sorting is a very efficient DFSORT/VSE sorting technique



FLR Incore Dataspace Sorting
R4 vs R3 Perf. Comparisons

DFSORT/VSE 3.4



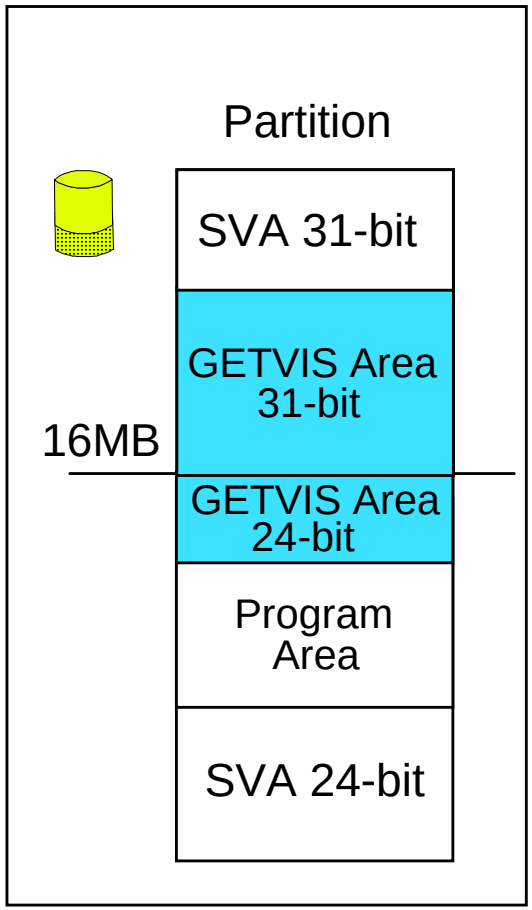
Improve your R4 performance by using optimal EXEC SIZE values

Getvis Sorting

DFSORT/VSE 3.4

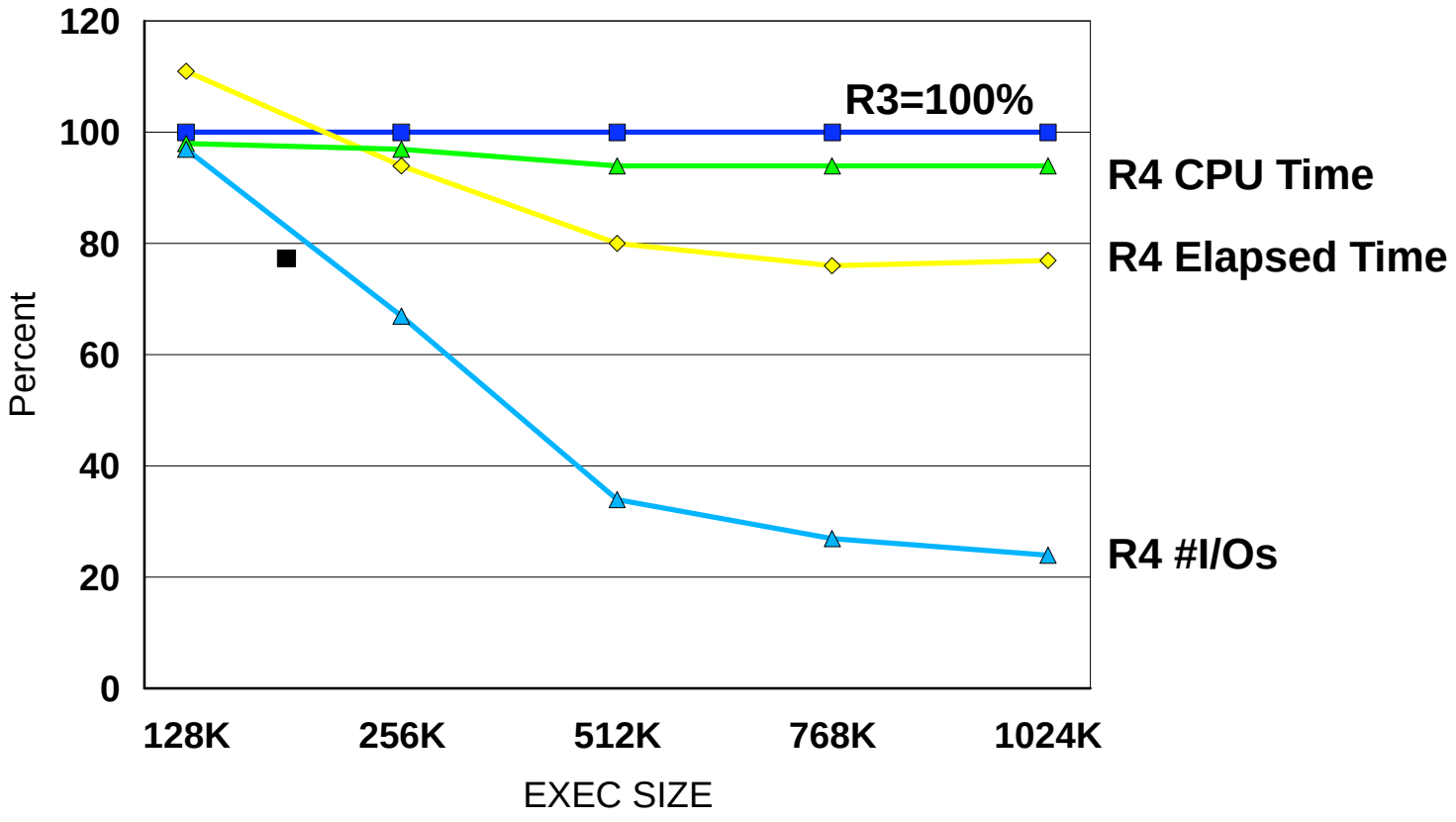
Getvis sorting is used if GVSIZE=n and DSPSIZE=0

Getvis sorting is another very efficient DFSORT/VSE sorting technique



FLR Incore Getvis Sorting
R4 vs R3 Perf. Comparisons

DFSORT/VSE 3.4



– **BAM sortin/sortout files used**

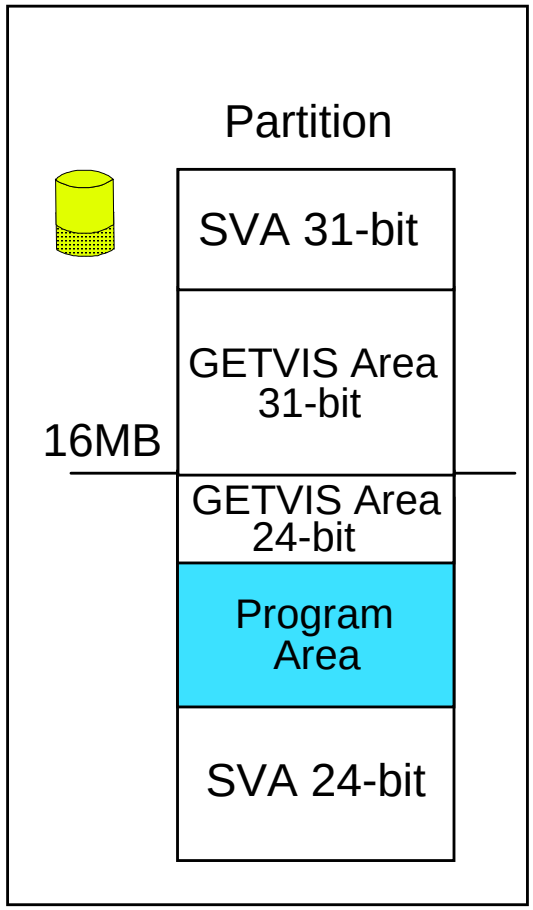
Improve your R4 performance by using optimal EXEC SIZE values

Partition Sorting

DFSORT/VSE 3.4

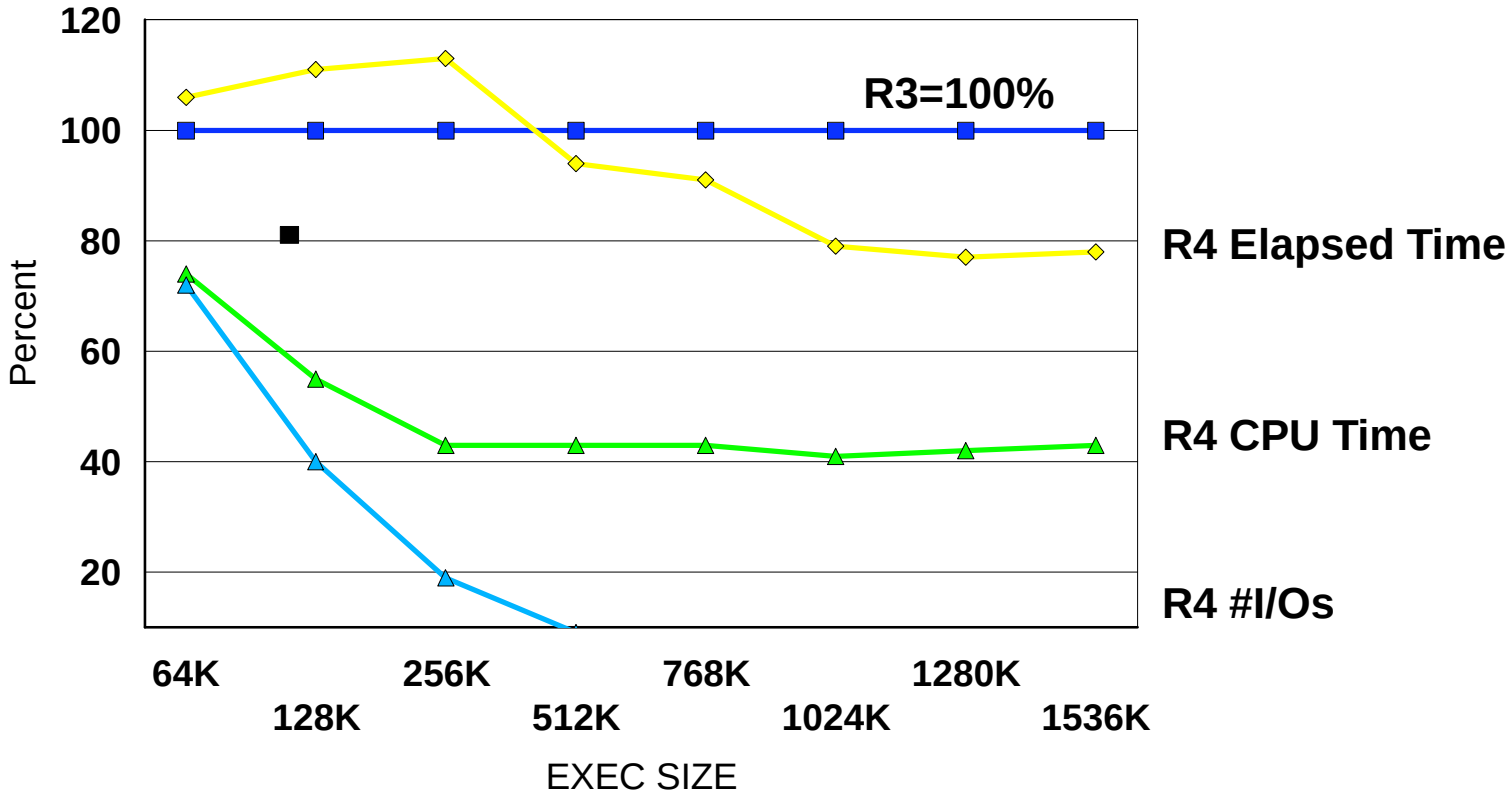
Program area sorting is the default if GVSIZE=0 and DSPSIZE=0

Program area sorting is the least efficient DFSORT/VSE sorting technique



FLR Copy
R4 vs R3 Perf. Comparisons

DFSORT/VSE 3.4

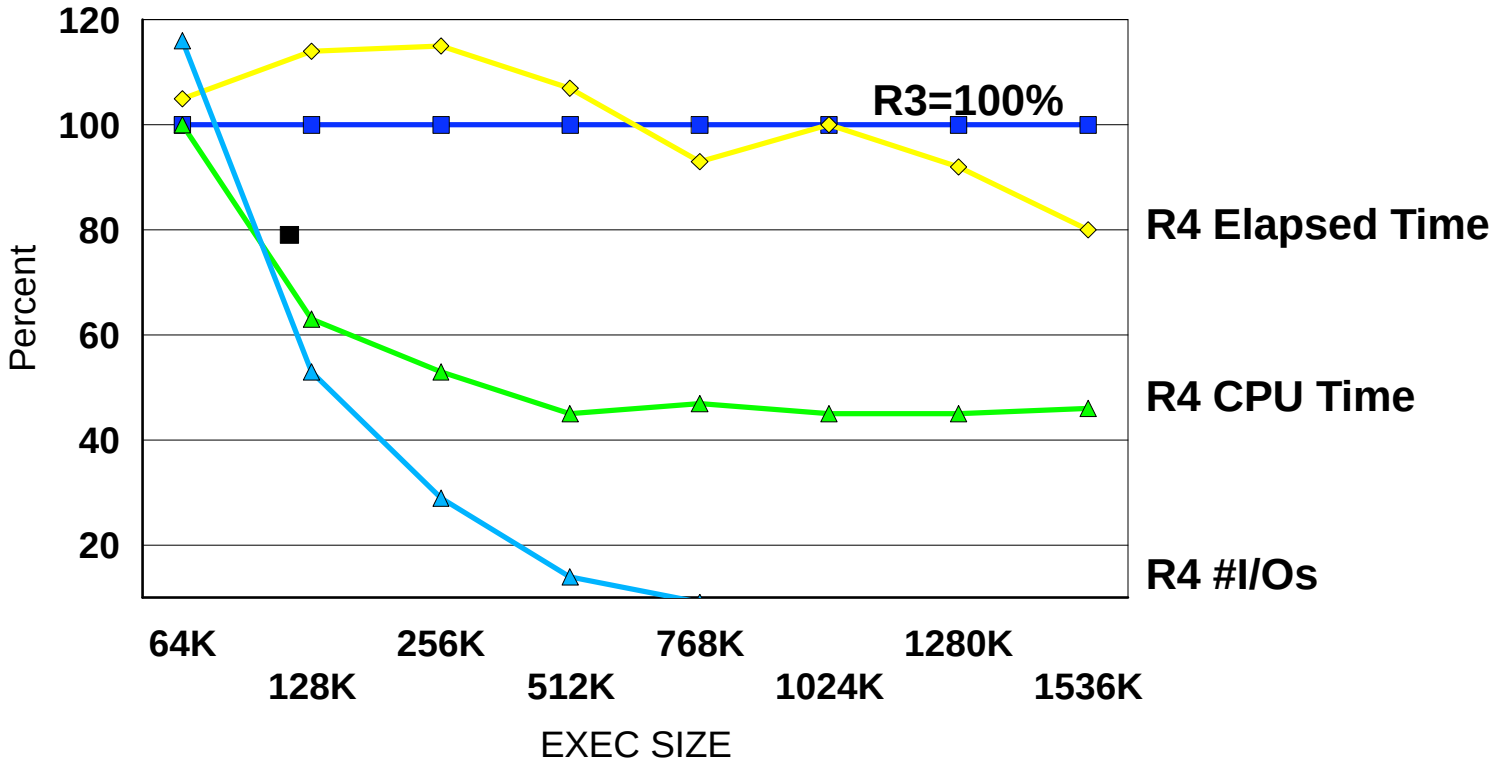


– BAM sortin/sortout files used

Improve your R4 performance by using optimal EXEC SIZE values

FLR Merge
R4 vs R3 Perf. Comparisons

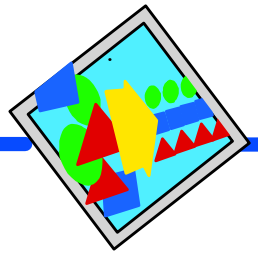
DFSORT/VSE 3.4



– BAM sortin/sortout files used

Improve your R4 performance by using optimal EXEC SIZE values

Factors Influencing Performance



DFSORT/VSE 3.4

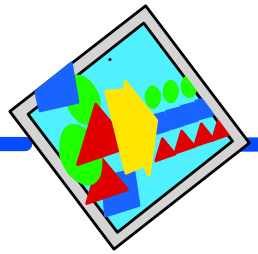
- Job specific
  - Amount of data to be sorted
  - Type of data (SAM/VSAM, VLR/FLR) and its location on I/O devices
  - Type of sorting (dataspace/getvis/partition sorting, incore/non-incore)

- Environment specific
  - VSE/ESA release (very slightly)
  - DFSORT/VSE release (use the latest release!)
  - DFSORT/VSE parameters used (install time/run time)
  - Available VSE sizes for partition, dataspace (essential area for sorting)

  - Speed of processor
  - Type and configuration of I/O subsystem

  - Native vs. VM/ESA Guest
  - Concurrent activities in VSE
  - VSE priority of DFSORT/VSE work

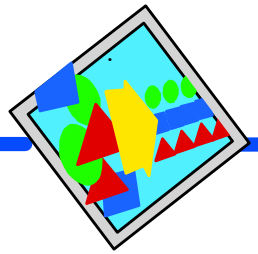
Storage Considerations
EXEC SORT,SIZE=xK



DFSORT/VSE 3.4

- DFSORT/VSE can run in 32KB of partition program area (meaning EXEC SORT,SIZE=32K) but this is NOT recommended.
- Start with EXEC SORT,SIZE=768K
- Increase the EXEC SIZE parameter value if DFSORT/VSE phases are not in the SVA or if you have:
  - Large input files
  - Spanned records
  - Very large blocks or logical records
  - User exit routines
  - Additional DFSORT/VSE functions (INCLUDE, OMIT, SUM, etc.)

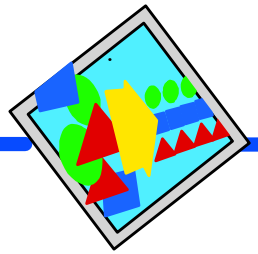
How to Set Up Efficient Sorts In A Nut Shell



DFSORT/VSE 3.4

1. Use getvis sorting or dataspace sorting. They both provide about the same performance. Note: only use DSPSIZE/GVSIZE=MAX when necessary or when concurrent workload is under control.
2. Try to provide a large enough GETVIS area or data space so that all the records can be sorted incore, meaning without using external work space (rule of thumb, provide a GETVIS area or data space as large as the file size).
3. Give DFSORT/VSE plenty of partition program area so it can have enough room for efficient buffering. In general, EXEC SORT,SIZE=768K, is good.
4. Make sure the DFSORT/VSE STORAGE installation and/or run-time option is set to no less than 768K.

Additional Capabilities and Enhancements



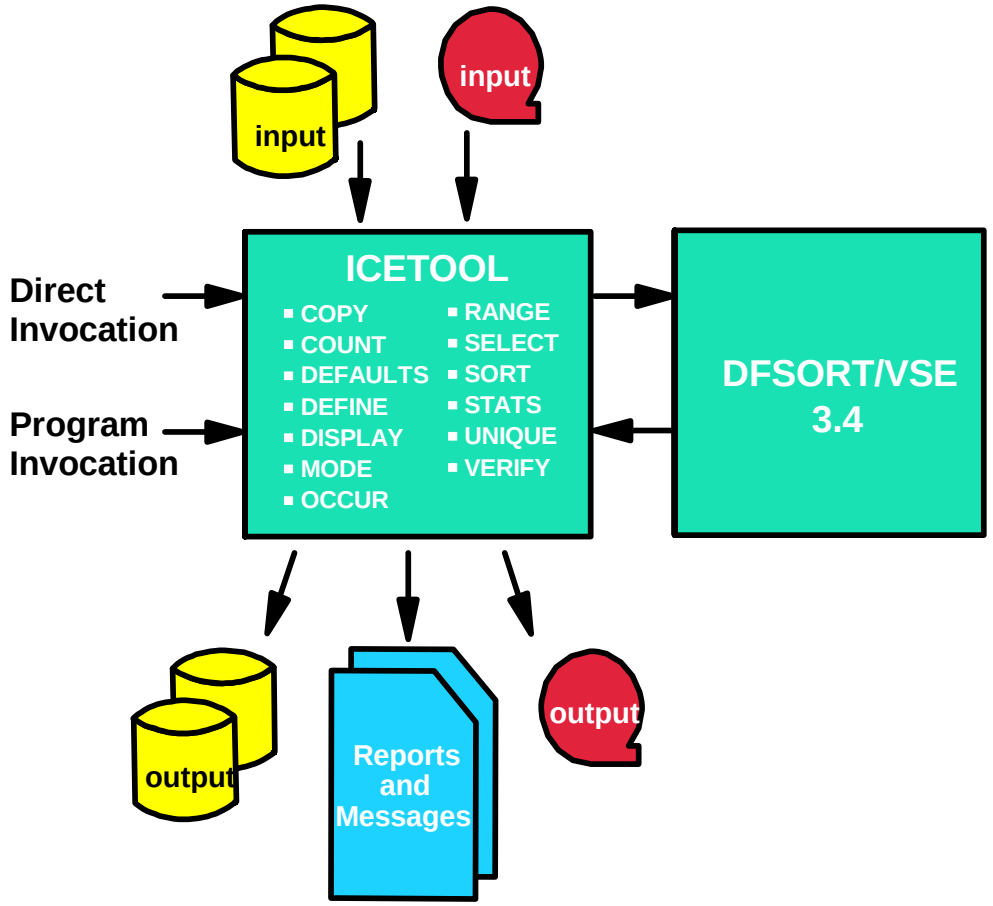
DFSORT/VSE 3.4

- ICETOOL Utility
- Online Message Explanations
- OUTREC Enhancements
- INCLUDE/OMIT Enhancements
- File Management System Enhancements
- STXIT Routine for Abend Recovery
- Additional Enhancements

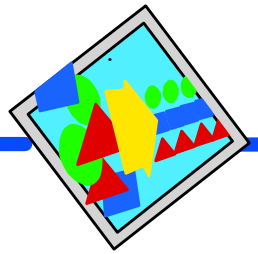
What is ICETOOL?

DFSORT/VSE 3.4

ICETOOL is a multi-purpose DFSORT/VSE utility



ICETOOL DEFAULTS



DFSORT/VSE 3.4

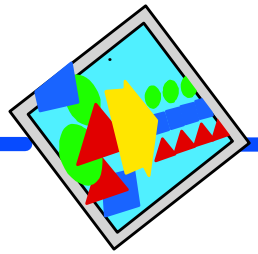
*** DEFAULTS - prints the DFSORT/VSE installation defaults.**
*** Example: print the defaults to SYSLST.**
DEFAULTS LIST(LST)

```
DFSORT/VSE INSTALLATION (ILUINST) DEFAULTS            - 1 -

* ONLY SHOWN IF DIFFERENT FROM THE SPECIFIED INSTALLATION DEFAULT

PARAMETER     INSTALLATION DEFAULT     IBM-SUPPLIED DEFAULT *
-----------------  ---------------------------------  ---------------------------------
CHALT          NOCHALT
DIAG           DIAG                          NODIAG
DUMP           NODUMP
EQUALS         NOEQUALS
ERASE          NOERASE
 .
 .
 .
```

ICETOOL
Reporting Made Easy

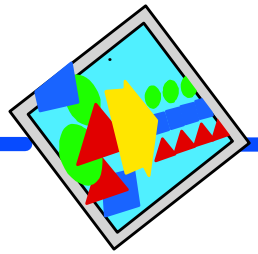


DFSORT/VSE 3.4

Population Summary:

```
// JOB EXAMP JOBB,PROGRAMMER
// LIBDEF PHASE,SEARCH=(PRD2.PRSMPROD)
// DLBL VSESPUC,'VSESP.USER.CATALOG',,VSAM
// DLBL IN2,'SORT.COUNTRY',,VSAM,DISP=(OLD,KEEP),
          CAT=VSESPUC
// EXEC ICETOOL,SIZE=768K
  DEFINE NAME(COUNTRY) TYPE(F) LENGTH(70)
  SORT FROM(COUNTRY) TO(TEMP1)
    USTART
       SORT FIELDS=(41,20,CH,A)
    UEND
*  Print a report
  DISPLAY FROM(TEMP1) LIST(012) -
     TITLE('Population Summary') DATE(MD4/) -
     HEADER('Country') HEADER('Population') -
     ON(41,20,CH) ON(61,10,ZD,A1) -AVERAGE('Average Population:') -
     TOTAL('Total Population:')
/*
/&
```

ICETOOL
Reporting Made Easy



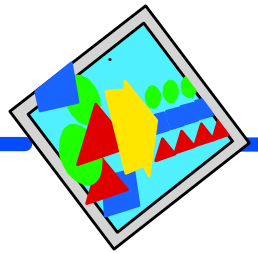
DFSORT/VSE 3.4

ICETOOL Output in SYS012:

Population Summary **5/22/2000**

Country Name	Population
Afghanistan	16,900,000
Austria	10,000,000
Brazil	147,400,000
Egypt	54,800,000
Namibia	12,800,000
Nepal	18,700,000
Nigeria	115,300,000
Panama	2,500,000
Peru	21,400,000
Portugal	10,400,000
San Marino	170,000
Senegal	7,200,000
Average Population:	34,797,500
Total Population:	417,570,000

Online Message Explanations (OME)

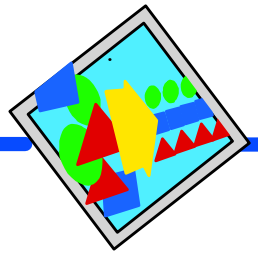


DFSORT/VSE 3.4

- Explanations of DFSORT/VSE messages can be displayed on the console
- Can be used similar to VSE/ESA OME

Diagnose and correct errors more quickly

OUTREC Enhancements



DFSORT/VSE 3.4

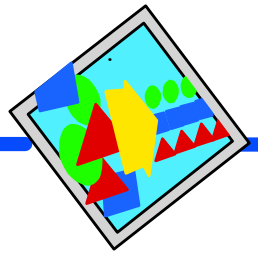
New OUTREC Editing Masks for Numeric Data

```
OUTREC FIELDS=(5:21,8,ZD,M12,
               25:46,10,ZD,M19)
```

Mask	Pattern	Value	Result
M12	SIII,III,III,III,IIT	-0012345	-12,345
M19	SI.III.III.III.IIT,TT	+0012345	1.234,56

Allows you to format your reports

OUTREC Enhancements



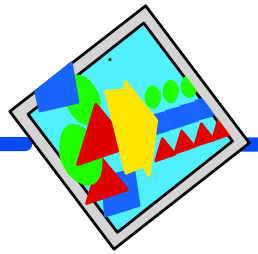
DFSORT/VSE 3.4

Lookup and Change

```
OUTREC FIELDS=(1,4,10,2,
       CHANGE=(15,
                   C'CA',C'CALIFORNIA',
                   C'NY',C'NEW YORK',
                   C'SD',C'SOUTH DAKOTA',
                   C'AZ',C'ARIZONA'),
       NOMATCH=(C'*INVALID ENTRY*'))
```

Improves productivity

INCLUDE/OMIT Capabilities



DFSORT/VSE 3.4

INCLUDE and OMIT Substring Search

Before

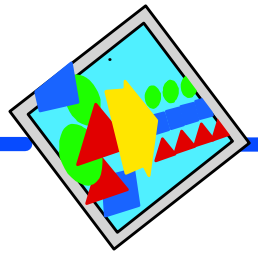
```
INCLUDE  COND=(11,2,EQ,C'OK' ,OR,
               12,2,EQ,C'OK',OR,
               13,2,EQ,C'OK',OR,
               21,3,EQ,C'J69',OR,
               21,3,EQ,C'J82',OR,
               21,3,EQ,C'L92')
```

After

```
INCLUDE  FORMAT=SS,
         COND=((11,4,EQ,C'OK' ),OR,
               (21,3,EQ,C'J69,J82,L92'))
```

Improves productivity

INCLUDE/OMIT Capabilities



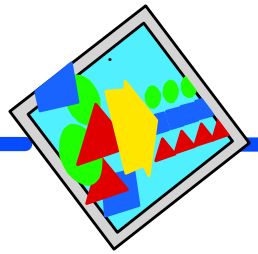
DFSORT/VSE 3.4

INCLUDE/OMIT Bit Level Logic

- **Bit operator with hex or bit mask**
  Example: (5,1,BI,SOME,X'C2')
- **Bit comparison with bit constant**
  Example: (2,1,BI,EQ,B'01..1...')

Create subsets of records based on flag fields

INCLUDE/OMIT Enhancements



DFSORT/VSE 3.4

More INCLUDE/OMIT conditions

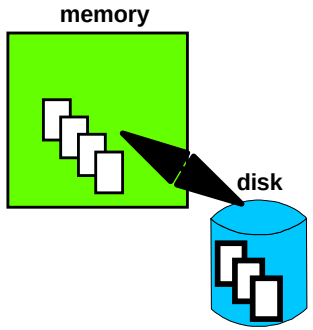
You can now:

- **Use a significantly larger number of INCLUDE/OMIT conditions**

Additional INCLUDE/OMIT capabilities

Interaction with File Management Systems

DFSORT/VSE 3.4



- R3/R4 Initial FMS Support

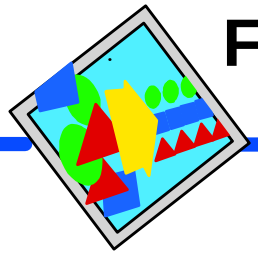
Input, output and work files don't need to be completely defined.

Enhanced

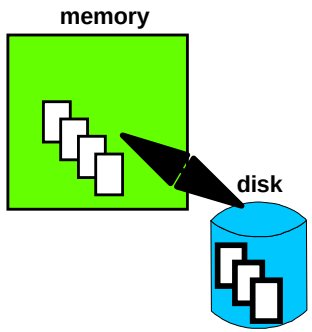
- R4 PQ19271 FMS Enhancement

INPFIL/OUTFIL BLKSIZE doesn't need to be specified. We'll use the BLKSIZE returned from the FMS.

Interaction with File Management Systems, cont.



DFSORT/VSE 3.4

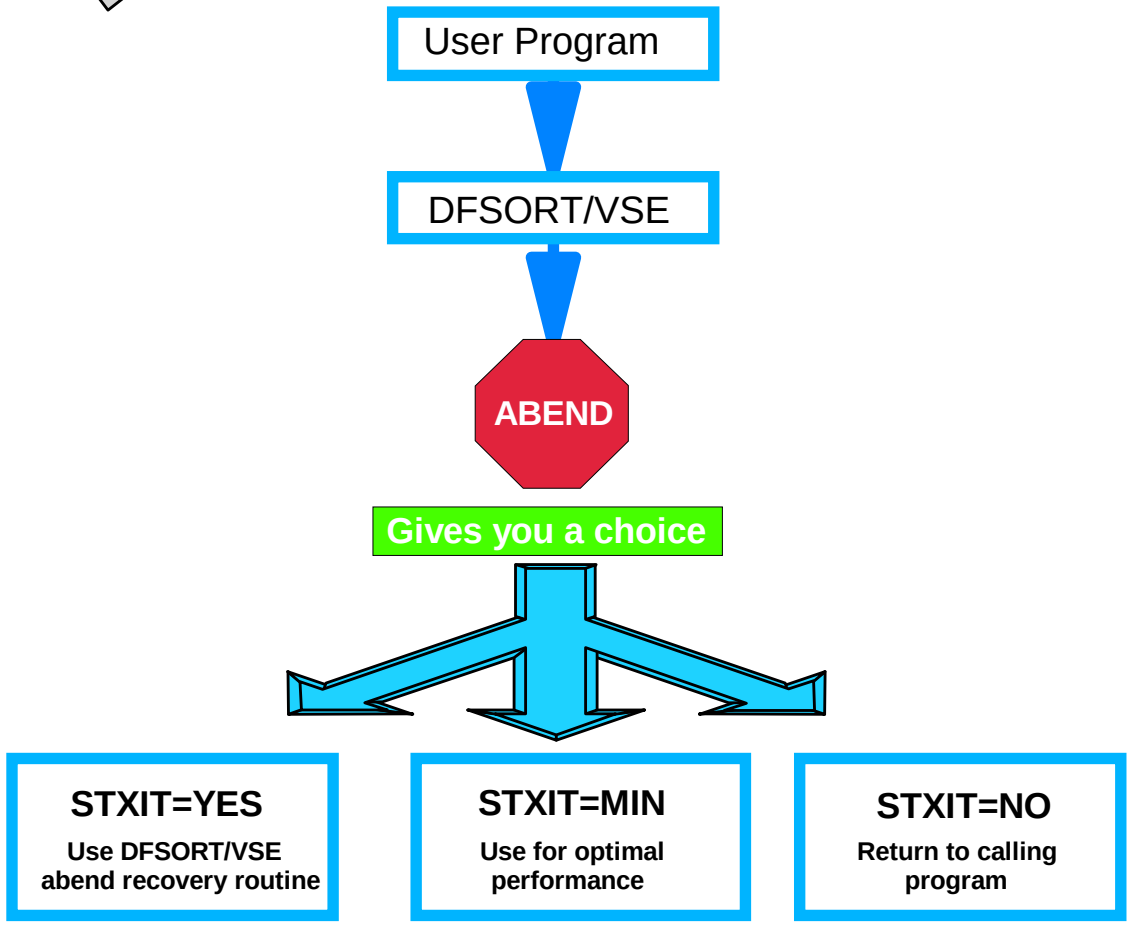


With PQ19271, our FMS support Provides:

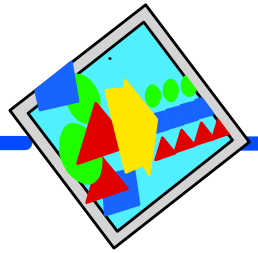
- Dynamic logical and physical device assignment
- Dynamic primary and secondary extent allocation
- Dynamic block size assignment for input and output files
- Allocation of a secondary extent dynamically when the primary work extent is exhausted
- Truncation of output files
- Closing of work files

STXIT Routine for Abend Recovery

DFSORT/VSE 3.4



STXIT
What setting should you use?



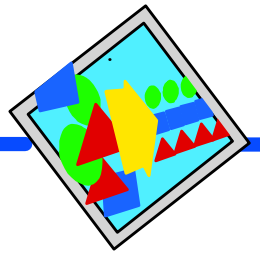
DFSORT/VSE 3.4

If you use LE then use the following chart:

E15/E35 User Exit Routines	R4	<R4
yes	STXIT=MIN	STXIT=NO
no	STXIT=MIN|YES	STXIT=YES
IBM Default	STXIT=MIN	STXIT=YES

- **STXIT=YES** allows DFSORT/VSE to do abend recovery and cleanup if an abend occurs.
- **STXIT=MIN** was introduced in DFSORT/VSE R4. If no user exit routines are used, it's the same as STXIT=YES. If a user exit routine is used and an abend occurs in the user exit routine, DFSORT/VSE will not perform abend recovery; DFSORT/VSE will not restore it's STXIT.
- **STXIT=NO** turns off DFSORT/VSE abend recovery. In R2 and R3, it also disables DFSORT/VSE's recovery feature for SAM ESDS multivolume work files.

Additional Enhancements



DFSORT/VSE 3.4

OPTION PRINT=CRITPLUS (APAR PQ17888)

Same as PRINT=CRITICAL plus one of the following msgs:

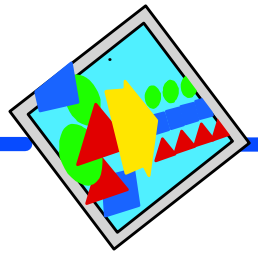
- ILU321I {SORT¦MERGE¦COPY} COMPLETE, INSERT v, DELETE x, IN y, OUT z
- ILU323I {SORT¦MERGE¦COPY} COMPLETE, IN y, OUT z
- ILU333I {SORT¦MERGE¦COPY} ERROR, IN y, OUT z

ENDEOD=YES | NO (APAR PQ19271)

- Specifies how DFSORT/VSE processes statements following the END control statement.
- YES requires /* (end-of-data) and follows the system requirement
- NO does not require /* (is equivalent to the way S/M 2.5 and R1 handled it)

APARs PQ17888 & PQ19271

Future Enhancements

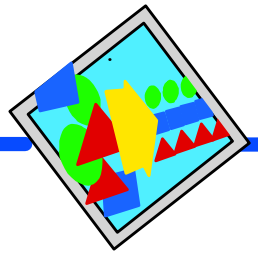


DFSORT/VSE 3.4

- **Enable D/V to sort very large files (240 gig file)**
- **WAVV Requirement**

Hope to provide this by YE2000

Year 2000 Features



DFSORT/VSE 3.4

- **Year 2000 Features**
- **COBOL Year 2000 Features**

Year 2000 Features

DFSORT/VSE 3.4

Evolution of DFSORT/VSE's Y2K features

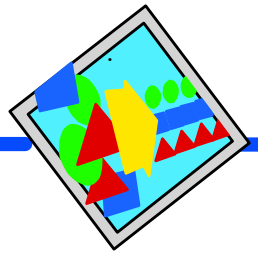
- First generation requires V3R4, V3R3, or V3R2 with PTF UN99635
- Second generation requires V3R4 with UQ33592 and UQ35657

DFSORT's Century Windows

DFSORT/VSE 3.4

- **Fixed Century Window**
  - "Fixed" because its range doesn't change
  - Y2PAST=1989 gives a CW of 1989-2088
  - Example: Dates for a company started in 1989
- **Sliding Century Window**
  - "Sliding" because its range changes every year
  - Y2PAST=8 gives a CW of 1989-2088 in 1997, 1990-2089 in 1998, and so on
  - Example: Dates for a sliding 30-year average
- **Y2PAST** is an installation and run time option

Year 2000 Features
First Generation



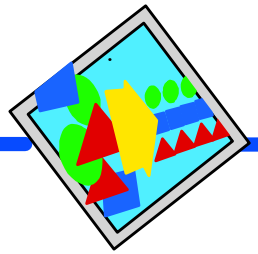
DFSORT/VSE 3.4

First Generation - Year Formats (12/96)

- Y2C, Y2Z, Y2P, Y2D
- Y2PAST for century window
- Y2S, Y2B added in R4

- Support not as easy to use as it could be
- Must split out year from month and day
- No INCLUDE/OMIT capability
- Conversion to CH, but not to PD

Year 2000 Features
Second Generation



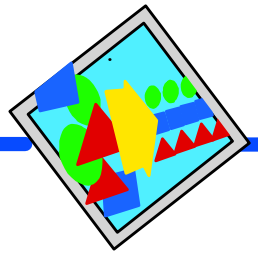
DFSORT/VSE 3.4

New Generation - Full Date Formats (8/99)

- **Y2T, Y2U, Y2V, Y2W, Y2X, Y2Y**
- **Y2PAST for century window**

- **Handles yyx...x and x...xyy dates**
- **Handles special indicators**
- **Provides full INCLUDE/OMIT capability**
- **Conversion to CH w/ and w/o separators**
- **Conversion to PD**
- **APAR: PQ22126 and PQ30735**

Year 2000 Features
COBOL



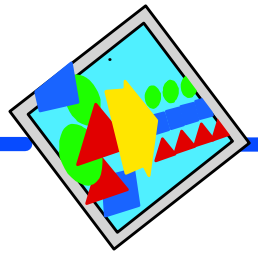
DFSORT/VSE 3.4

Sort and Merge with COBOL

- **COBOL MLE - automatic**
  - Use DATE FORMAT clause
  - Builds statements with DFSORT's Y2x format
  - APAR PQ22020 - upper case English
  - APAR PQ22024 - LE runtime

- **Without COBOL MLE - explicit**
  - Use SYSIPT to pass SORT statement with Y2K formats

COBOL and SORTWK Files

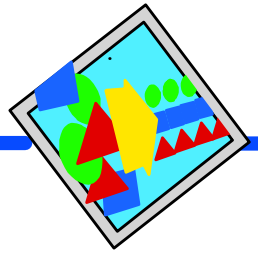


DFSORT/VSE 3.4

Optimizing DFSORT/VSE's performance with COBOL when sort work files are needed.

- DFSORT/VSE will open all the sort work files that are specified in SORT...WORK=n
- COBOL/VSE and COBOL II dynamically generate the WORK=n parameter on the SORT statement at run-time based on the DLBLs in the JCL. DOS/VS COBOL does not.
- COBOL/VSE uses the DLBL information in the following way (highest priority to lowest priority):
  - JCL of the job (SORTWKn DLBLs)
  - Partition standard labels
  - System standard labels

COBOL and SORTWK Files Recommendations



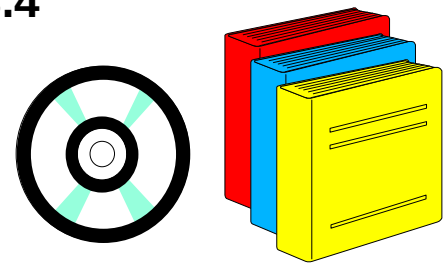
DFSORT/VSE 3.4

- Use WORK=n values as small as possible. Use dataspace sorting or getvis sorting to reduce the need to use work files.
- Remove any unnecessary SORTWK DLBLs to avoid any unnecessary opens (in the job or standard labels).
- Use SAM ESDS work files when possible. DFSORT/VSE will only allocate/open the first extent. Be sure to specify:
  - A good size secondary extent
  - WRKSEC

Note: The allocation of work files does not significantly effect performance since if no work files are needed, they will not be used.

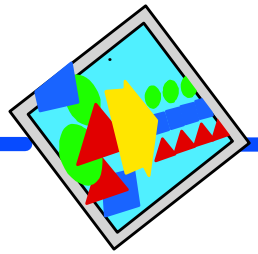
DFSORT/VSE Library

DFSORT/VSE 3.4



- **Publications**
  - General Information (GC26-7039)
  - Application Programming Guide (SC26-7040)
  - Messages, Codes and Diagnosis (SC26-7132)
  - Reference Summary (SX26-6008)
  - Installation and Tuning Guide (SC26-7041)
  - Diagnosis Guide (SY27-7600)
  - Getting Started with DFSORT/VSE (SC26-7101)
- **DFSORT/VSE Library (SBOF-6130)**
- **IBM Online Library VSE Collection (SK2T-0060), CDROM**
- **All DFSORT/VSE publications are available on the web!**

Summary



DFSORT/VSE 3.4

- **Existing Capabilities Enhancements**
  - Dataspace sorting
  - Getvis sorting
  - Partition sorting
  - ICETOOL
- **New Features**
  - Online Message Explanations (OME)
  - OUTREC enhancements
  - INLUDE/OMIT enhancements
  - STXIT enhancements
- **Year 2000 features**