

SyncSort for z/OS Services

A Partnership for Excellence

Service Offerings

Professional Services

• Training Services

Support Services



Training Services

Getting the Most from SyncSort for z/OS

A 3-hour Training Course for Application Developers

Why write programs when SyncSort control statements are easier to code, easier to maintain, and produce applications that run faster?

Beyond Sorting

Yes, SyncSort can tear through sort applications at lightning speed. But did you know that many sort applications could do much more, in less time, and with less programming effort? Learn how with the SyncSort course for application programmers, *Getting the Most from SyncSort for z/OS*.

What Customers Say...

*"Awesome! As a result of the course, we were able to get a **2-hour** job down to **20 minutes** to fit in a critical window.... Everyone learned features they didn't know about."*

- Sheldon Lane, IT Training Coordinator,
Verizon Wireless

*"After the class we were able to streamline SyncSort code substantially. The job ran **much faster**, and everything downstream ran faster. The entire stream savings was **45 minutes**."*

- Ronald Rozek, Team Lead, Technical Systems,
Thomson Reuters

Get the Most from SyncSort

Getting the Most is taught by a SyncSort analyst with extensive experience helping customers one-on-one to exploit the full range of SyncSort features and functions. Now that experience is packed into a 3-hour presentation and discussion forum. Attendees will learn how to:

- **Speed application development.** Develop applications in a fraction of the time required for home-grown programs, and spend less time maintaining them. Save hundreds of hours of programming time, year after year.
- **Speed processing.** Utilize SyncSort features to replace inefficient applications that restrict system throughput or that can't finish within critical batch processing windows.
- **Do more in less time!**



Getting the Most from SyncSort for z/OS

Course Outline

Exploit SyncSort Capabilities

SyncSort capabilities are well documented – they're not hidden. But programmers reach for the tools they know. If they don't know how SyncSort can save them time, they'll go to the inefficient methods they're familiar with, such as COBOL coding, C programming, report generators, and multiple job steps.

Getting the Most from SyncSort for z/OS will help programmers:

- Use SyncSort JOIN to combine data from two independent input files based on matching or unmatching key values; identify and write unique records that exist in one of the two input files; SyncSort JOIN is as flexible as DB2 or Access JOIN but works on mainframe flat files
- Eliminate equally-keyed records and write deleted records to a file
- Sum numeric data – highly efficient for large aggregation tasks
- Identify different values in a specific field within a file, enabling you to easily verify file contents
- Select only the records you really need for your application and process them conditionally
- Write complete reports – with headers, trailers, sections, totals, and counts – more efficiently than with a report program
- Perform data conversion
- Efficiently create multiple, custom output files in one pass of your data
- Dramatically reduce SORTWK DASD requirements, freeing the DASD pool for other applications

Course Details

- 3-hour presentation and discussion at customer site
- Optimum attendance is 15; for more people an additional course can be given on the same day
- Recommended prerequisites: some familiarity with JCL and basic SyncSort control statements
- Course materials: presentation printout, plus flash drive packed with documentation, white papers, best practices, and effective tutorials to get programmers up to speed fast with SyncSort features

*In just three hours,
programmers will be
able to use SyncSort to
get **more done** in
less time*

**For more information on
*Getting the Most from
SyncSort for z/OS*
contact your
SyncSort representative
at 201-930-8200**



Getting the Most from SyncSort for z/OS

Course Description

General

Purpose: Introduce application programmers to powerful capabilities of SyncSort for z/OS

Benefit:

- Reduce application development and maintenance time
- Develop applications that run faster while minimizing system impact

Length: 3 hours; can be given twice in a day

Optimum number of attendees: 15

Recommended prerequisites: Some familiarity with JCL and basic SyncSort control statements

Course materials

- Presentation printout
- Flash drive containing:
 - Complete SyncSort documentation
 - Whitepaper on exploiting zIIP, MIDAW, and PAV for performance gains
 - Best practices
 - Case studies
 - Tutorial booklets on JOIN, Data Utilities, SortWriter, MAXSORT

Syllabus

The course has 14 sections covering 12 SyncSort functions:

- Introduction
- JOIN Processing
- Record Selection
- DUPKEYS
- Data Manipulation
- UTFIL
- Visual SyncSort
- Dynamic Allocation
- MAXSORT
- PARASORT
- PAV Technology
- PROC SYNCSORT
- PipeSort
- Conclusion



Getting the Most from SyncSort for z/OS

Course Description

Introduction

- Main functions: sort, merge, copy
- Flow of the sort: relation of sort to input/output processing

JOIN Processing

- Equivalent to DB2 or MS Access JOIN
- Two JOIN input files specified in JCL
- Identify paired and unpaired (unmatching) key values
- Description of types of JOIN processing
- A simpler solution to a common application assignment
- SyncSort control statements for JOIN processing
- Possible JOIN outcomes

Record Selection

- Count records to start after or stop after: SKIPREC and STOPAFT, STARTREC and ENDREC
- INCLUDE and OMIT: JOIN, Boolean comparisons
- Special cases: bit-level, DATE, substring
- Performance issues
- UTFIL for output record selection: INCLUDE/OMIT, STARTREC/ENDREC, SPLIT/SPLITBY, SAMPLE, SAVE
- DUPKEYS
- Combine equally-keyed records: SUM, AVG, MIN, MAX, NONE
- Relationship to SUM
- Capture records deleted by DUPKEYS (XDUP/XSUM)
- Retain one of equally-keyed records
- Deal with SUM overflow

Data Manipulation

- Select only fields needed: improves performance
- Add new fields: character, hex/numeric values, date/time, sequence number, field expansion to avoid sum overflows
- Convert existing fields: numeric format, edit masks, CHANGE, ASCII to/from EBCDIC, uppercase to/from lowercase, alternate collating sequence, operators, conditional processing
- Convert RECFM

UTFIL

- Generate multiple output files with one sort step
- Specify unique record selection, data manipulation, DCB characteristics for each output file
- Generate reports efficiently with SortWriter



Getting the Most from SyncSort for z/OS

Course Description

Visual SyncSort

- Use Visual SyncSort to create and maintain complex SyncSort applications graphically from a PC
- Generate error-free control statements without any knowledge of SyncSort syntax

Dynamic Allocation

- Code SORTWK DDs to avoid "Sort Capacity Exceeded" conditions
- SORTWK allocation: how many, how large
- SyncSort DYNALLOC versus JCL-allocated SORTWKs

MAXSORT

- Run large jobs when sufficient DASD is not available

PARASORT

- Exploit parallel input processing to reduce elapsed time for multivolume and concatenated tape data sets

PAV Technology

- Unique exploitation of PAV to improve throughput
- PARASORT parallel processing applied to DASD

PROC SYNCSORT

- Replace PROC SORT in SAS® applications for big performance gains

PipeSort

- Run multiple, unique sorts simultaneously on a single pass of the input file for up to 50% elapsed time savings

Questions

**For more information on
*Getting the Most from
SyncSort for z/OS*
contact your
SyncSort representative
at 201-930-8200**