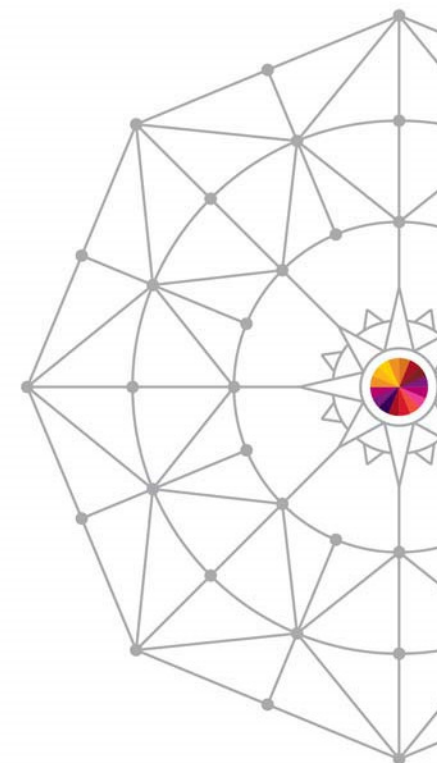


z/OS V2 R1 Sysprog Goody Bag

Session 16088

Bob Rogers

*IBM Distinguished Engineer, Retired
Trident Services, Inc.*



Insert
Custom
Session
QR if
Desired.



SHARE is an independent volunteer-run information technology association
that provides **education, professional networking and industry influence.**

Abstract

- After covering 2 versions of MVS/ESA, 10 releases of OS/390 and 13 releases of z/OS version 1, the famous goody bag continues into a new version of z/OS with Bob Rogers back again to educate and entertain us. z/OS 2.1 is the first z/OS release on the two year cadence so Bob will have no problem filling the session with goodies delivered in the release. But there still should be time for digressions into history and mainframe lore.

Trademarks

The following are trademarks of the International Business Machines Corporation in the United States, other countries, or both.

Not all common law marks used by IBM are listed on this page. Failure of a mark to appear does not mean that IBM does not use the mark nor does it mean that the product is not actively marketed or is not significant within its relevant market.

Those trademarks followed by ® are registered trademarks of IBM in the United States; all others are trademarks or common law marks of IBM in the United States.

For a complete list of IBM Trademarks, see www.ibm.com/legal/copytrade.shtml:

*, AS/400®, e business (logo)®, DBE, ESCO, eServer, FICON, IBM®, IBM (logo)®, iSeries®, MVS, OS/390®, pSeries®, RS/6000®, S/30, VM/ESA®, VSE/ESA, WebSphere®, xSeries®, z/OS®, zSeries®, z/VM®, System i, System i5, System p, System p5, System x, System z, System z9®, BladeCenter®

The following are trademarks or registered trademarks of other companies.

Adobe, the Adobe logo, PostScript, and the PostScript logo are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, and/or other countries.

Cell Broadband Engine is a trademark of Sony Computer Entertainment, Inc. in the United States, other countries, or both and is used under license therefrom.

Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

Intel, Intel logo, Intel Inside, Intel Inside logo, Intel Centrino, Intel Centrino logo, Celeron, Intel Xeon, Intel SpeedStep, Itanium, and Pentium are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

ITIL is a registered trademark, and a registered community trademark of the Office of Government Commerce, and is registered in the U.S. Patent and Trademark Office.

IT Infrastructure Library is a registered trademark of the Central Computer and Telecommunications Agency, which is now part of the Office of Government Commerce.

* All other products may be trademarks or registered trademarks of their respective companies.

List of Topics

- Zero Address Detection (ZAD)
- MACHMIG statement in LOADxx
- Instruction Counts in SMF Type 30 Record
- z/OS Software Counters from HIS
- Concatenation of VLF Parmlib Members
- Changing VLF Configuration without Restart
- JES-Independent JCL
- JES2 JCL Interpretation Right after Conversion
- Controlling Where JES2 Converts a Job
- Parallel Batch Recall
- SYSDSN ENQ Downgrade in JCL
- Symbol Substitution in Instream Data Sets
- Greater than 100 character Parms in JCL
- JES3 Supports Multiple Simultaneous Logons
- JES3 In-stream Data in PROCs and INCLUDEs
- Other JES3 Enhancements
- FORCE TCB Command
- System Symbol for z/OS release
- Reserve SVC and still get Fxx Abend
- Tracing DIES in System Trace
- Support for Time in the 2nd Epoch
- Some New Health Checker Routines
- Dynamically Add and Delete MCS Consoles
- System symbol updates for local system
- Line Comment in BCP Parmlib Members

Zero Address Detection (ZAD)

- ZAD is a new PER capability introduced on the z196.
 - A Zero Address Detection (ZAD) event is caused by the execution of an instruction that accesses (stores or fetches) storage using an operand address formed from a general register containing zero.
 - Detection only occurs when the PSW PER bit is on and ZAD is enabled.
- It is turned on by setting a SLIP PER trap of type ZAD.
 - A ZAD trap can help to detect errors that involve unintentionally referencing the PSA (aka Low Core). All instructions are monitored for ZAD events but you can limit ZAD events by limiting the address spaces and/or jobs for which PER is active.
- However, it is possible to limit the ZAD events that match for a SLIP trap by using other SLIP filtering keywords.
- ZAD is intended for use by customers and ISVs to diagnose their own problems in their own code. Events in IBM code are not to be considered errors and therefore should not to be reported to IBM L2.
 - There are many cases where a ZAD event is not an error.
 - Recommended use is with A=EXIT,AEXIT=IEAVTSZE. Read the book before using.

MACHMIG statement in LOADxx

- When migrating to a new machine with new facilities, it is often desirable to initially avoid using the new facilities – better to do some “compatibility mode” testing first.
- The MACHMIG statement in LOADxx is intended to facilitate this. It was introduced with OA38829 in z/OS 1.13
- Each operand identifies a facility that z/OS is to “hide” from software (at least software that is well-behaved and checks the facility list before using optional facilities)
- The operands supported for z/OS 2.1 are TX and EDAT2.
 - TX stands for transactional execution facility
 - EDAT2 is the enhanced-DAT-facility-2 (i.e., Flash)
 - One or more of the operands.
 - For example: MACHMIG TX,EDAT2 inhibits the use of transactional memory and EDAT2.

Instruction Counts in SMF Type 30 Record

- z/OS can now use hardware instrumentation data to include instruction counts in the SMF data for jobs.
- Instruction counts may be a more stable metric for CPU resource consumption than CPU time.
 - As processor designs change, some jobs benefit more than others, causing variability in the relative charge back costs based on CPU time.
 - There is also variability in CPU time caused by the use of shared processor resources like shared cache levels.
 - Future design changes are likely to aggravate the problem

http://www.ibmssystemsmag.com/mainframe/trends/z-os/SMT_challenges/

Instruction Counts in SMF Type 30 Record

- To get instruction counts in Counter Data Section, HIS must activate the CPU Measurement Facility to collect at least the basic counter set.
 - MODIFY HIS,BEGIN,CTRONLY,CTRSET=(B) is minimal
 - MODIFY HIS,BEGIN,CTRONLY,CTRSET=(B,E) is recommended
- SMFPRMxx must contain the keyword SMF30COUNT
- SMF 30 Counter Data Section Fields:
 - SMF30_Inst_CP_Task, SMF30_Inst_CP_NonPreemptSRB, SMF30_Inst_CP_PreemptSRB, SMF30_Inst_Offload, SMF30_Inst_OffloadOnCP, SMF30_Inst_CP_Enclave, SMF30_Inst_Offload_Enclave, SMF30_Inst_OffloadOnCP_Enclave, SMF30_Inst_CP_DepEnc, SMF30_Inst_Offload_DepEnc, SMF30_Inst_OffloadOnCP_DepEnc, SMF30_InstCaptDisruption, SMF30_InstCaptLimited
- If the counter data is not available, SMF30_InstCaptDisruption is set on in the record.

Instruction Counts in SMF Type 30 Records

- The new instruction count data can be mapped back to the CPU time data:
 - SMF30CPT is comparable to the sum of
 - SMF30_Inst_CP_Task
 - SMF30_Inst_CP_PreemptSRB
 - SMF30_Inst_OffloadOnCP
 - SMF30_Inst_CP_Enclave
 - SMF30_Inst_OffloadOnCP_Enclave
 - SMF30_Inst_CP_DepEnc
 - SMF30_Inst_OffloadOnCP_DepEnc
 - SMF30ASR is comparable to SMF30_Inst_CP_PreemptSRB
 - SMF30CPS is comparable to SMF30_Inst_CP_NonPreemptSRB

z/OS Software Counters from HIS

- The Enhanced Monitor Facility (EMF) introduced on z196 turns a monitor class into a “software driven counting facility”
 - z/OS uses this to count specific software events. For example:
 - Number of times a particular SVC is issued
 - Number of times a task or SRB is dispatched
 - Number of times a particular lock is obtained
 - Because of the use of Monitor Call to request counting, the overhead is near zero when not activate.
 - Because of EMF, the overhead is very low even when counting is software event active.
- HIS encapsulates these counts as the **ZOS Counter Set**
 - The HIS Profiler writes to the .CNT file and SMF Type 113 record, when requested via CTRSET=(ZOS).
 - *Currently, what each counter means is IBM internal only*

US Patent US8572357 B2

Monitoring events and incrementing counters associated therewith absent taking an interrupt

Complete your session evaluations online at www.SHARE.org/Pittsburgh-Eval

Concatenation of VLF Parmlib Members

- When VLF is started, up to 16 COFVLFxx parmlib members can be concatenated to form one VLF configuration
 - class definitions may span from one member to the next
- VLF treats the entire concatenation of Parmlib members as one configuration.
 - Duplicate definitions will be rejected with the appropriate existing error messages
- The command syntax is:
 - `s vlf,sub=mstr,nn=(aa,bb,cc)`

Changing VLF Configuration without Restart

- The VLF configuration can now be changed with a Modify command, without needing to restart VLF
 - Classes may be added or deleted.
 - Major names (EMAJ or EDSN) may be added to or deleted from an existing class.
 - MaxVirt can be specified, raised, or lowered for an existing class.
 - AlertAge can be specified, raised, or lowered for an existing class.
- The command is: `MODIFY VLF,REPLACE,NN=`
- The existing VLF cache is preserved to the extent that cached objects are still in the configuration

JES-Independent JCL

- Support new SYSTEM and SYSAFF keywords for the JOB statement in both JES3 and JES2.
- Reduces need to use JES-specific JECL
 - JES3 `/*MAIN SYSTEM=`
 - `SYSTEM=` or `SYSAFF=` keyword overrides `/*MAIN SYSTEM=` JECL statement.
 - JES2 `/*JOBPARM SYSAFF=`
 - `JOBPARM SYSAFF` ignored if JOB card `SYSTEM` or `SYSAFF` specified
- `SYSTEM` and `SYSAFF` on job card are mutually exclusive
 - Specifying both results in JCL error

JES2 JCL Interpretation Right after Conversion

- Conversion processing does the first pass at processing JCL
 - Deals with PROCs and INCLUDEs
 - Does basic parsing of data into “text units”
 - Currently run in a subtask in JES2 address space in conversion phase
- Interpretation processing creates control blocks for the initiator to run the job
 - Fully validates JCL (values of parameters)
 - Certain errors only detected by the interpreter
 - Currently runs in initiator when a job is selected
 - OUTPUT card processing cannot be done until the interpreter is run
 - Needs to run against control blocks interpreter creates
- OUTPUT cards (including JESDS= and MERGE=YES) do not apply if the job does not run.

JES2 JCL Interpretation Right after Conversion

- New option to run interpreter after conversion before job runs
 - If selected, both converter and interpreter run in a new address space
 - Address space name *jesxC1xx* where *jesx* subsystem name, *xx* is instance number
 - Both run during the conversion phase (i.e. before job selection)
 - When job is selected, interpreter is not run again
 - Control block are read from SPOOL and relocated
 - All systems in MAS must be z/OS 2.1 and \$ACTIVATE at LEVEL=Z11, otherwise the option has no effect.
- New options on JOBDEF (the scope is a member)
 - INTERPRET=INIT|JES – Selects to run interpreter under JES2 or INIT
 - CISUB_PER_AS=*nn* – indicates how many subtasks per JES2. The Default is 5 per address space

Controlling Where JES2 Converts a Job

- Prior to z/OS 2.1, conversion is controlled by factors like system affinity
- In z/OS 2.1 there is the ability to select conversion system based on SCHENV
 - A New CNVT_SCHENV option on JOBDEF (scope MAS)
 - CNVT_SCHENV=IGNORE|HONOR
 - Scheduling Environment parsed at input phase (or assigned)
 - HONOR says to only convert where SCHENV is available
 - If no system is available, the job waits to convert
 - Works on job that went through input on 2.1 and 2.1 converters
 - Down level members can select even if HONOR is set
 - That is, they do not honor HONOR

Parallel Batch Recall

- Serial Recall
 - Traditionally, Batch Allocation does a Catalog Locate for each data set
 - If any data sets are migrated, HSM is called to recall each data sets
 - When there are multiple data sets, the initiator waits for each one and is tied up not doing any productive work, resulting in delays in other jobs executing
- Parallel Recall
 - Request HSM recalls in parallel and reduce initiator idle time
- The option is enabled by a new ALLOCxx keyword
 - BATCH_RCLMIGDS(SERIAL) is legacy behavior and default
 - BATCH_RCLMIGDS(PARALLEL) requests the new capability
- This not the same as the JES3 function that requests recalls before the job is selected.

SYSDSN ENQ Downgrade in JCL

- Since time immemorial, if any step of a batch job allocates a data set for non-shared use, the SYSDSN ENQ is held exclusive for the entirety of the job, even if the final steps do not need exclusive control
 - This leads to other jobs that request the data set to wait until the ENQ holding job ends
- In z/OS 2.1, the SYSDSN ENQ can be downgraded to Share
 - This option can be requested on a per-job or per-jobclass basis with a new JOB statement keyword
 - DSENQSHR=<ALLOW | DISALLOW | USEJC>
 - Or on a JES Jobclass base
 - E.g. \$T JOBCLASS(<class>),DSENQSHR=< ALLOW | DISALLOW>
- Allows better parallelism of batch job execution when multiple jobs need to share a data set resource.

Symbol Substitution in Instream Data Sets

- z/OS 2.1 JES2 supports the use of symbols in the records of instream data sets.
 - This new feature allows application designers to use the same symbolic parameters in job's JCL and in job's instream data sets
- The substitution happens at the point in time when application reads a record from the instream data set
- The following types of symbols can be used:
 - JCL symbols – made available through the use of EXPORT statement
 - JES symbols – dynamically created by application
 - System symbols – MVS system symbols
- Substitution is controlled by a new SYMBOLS keyword on DD statement that defines the instream data set.
 - The default is not to perform symbol substitution.

There is more enhanced support for Symbols in V2.1 not covered here

Greater than 100 character Parms in JCL

- The new PARMDD= keyword on the EXEC JCL statement specifies the DDname of a data set or USS file.
- The data set can contain a string of up to 32K characters.
- If it is an instream data set it can contain symbols that are resolved by JES2.

– Here's an example of a DD * instream data set:

```
//D83RRR1A JOB MSGLEVEL=1
//S1          EXEC PGM=CHECKSMF,PARMDD=PARMINDD
//PARMINDD DD *
             CHECK SYSTEM=&SYSNAME
/*
```

- Of course, use of the PARMDD= keyword is mutually-exclusive with use of the PARM= keyword.

Greater than 100 character Params in JCL

How the records in the data set are processed:

- Input records with a fixed record format and records from JES2 instream data sets are examined for the presence of *sequence numbers*
 - 8 contiguous numeric characters in the last 8 bytes of a record
 - If sequence numbers are found, the record length is adjusted so that the sequence numbers are not seen in subsequent processing.
- Each input record is examined for trailing blanks.
 - If trailing blanks are found, the record length is adjusted so that the trailing blanks are not seen by subsequent processing
- Double ampersands in the parameter string are converted to single ampersands.
- The parameter string passed to the program is formed by the simple left to right concatenation of the input records after any sequence numbers and trailing blanks have been eliminated.

JES3 Supports Multiple Simultaneous Logons

- Standard TSO logon can be used on multiple systems in a sysplex with the same user id.
- When a TSO user is logged onto multiple systems, the following notification rules apply:
 - When a system was specified explicitly, then notification messages are sent to that system.
 - When no explicit specification is made, then the notification goes to the system from which the TSO user submitted the job.
 - When no system specification is found, then a search is made starting with the first system defined via MAINPROC; the first occurrence of the TSO user id triggers the notification to be sent to that system.
 - When no user is logged on, then the notification is sent to the global.
- A DUPLOGON inish keyword is added with APAR OA41998 and APAR OA42062 provides toleration support for down-level JES3 systems.

JES3 In-stream Data in PROCs and INCLUDEs

- JES3 now allows in-Stream data in JCL PROCs & INCLUDEs
 - Works for batch jobs as well as started tasks
- JCL Conversion requires z/OS 2.1 but the job can run on down-level system
- New SYSIN data sets are included in extended status DSLIST function
- New SYSIN data sets are visible via SDSF after toggling the 'Options' value for 'Change include SYSIN to Off/On'.
- New SYSIN data sets are NOT included in SPOOL Data Set Browse of JCLIN
 - The data sets were not part of original JCL submitted

Other JES3 Enhancements

- Dynamic spool delete
 - Allows spool deletion via the *F CONFIG command or hot start with refresh
 - Single Track Table data moved as part of *F Q,DD=ddn,DRAIN command
 - Dump job by DD name allows spool offload before spool deletion
- 8-character CLASS specification on JOB statement
- Job tracking via ENF 70
- Inish OPTIONS ...
 - DUMP – only PRDMP accepted
 - SDI - obsolete

FORCE TCB Command

- The FORCE command is enhanced to allow a directed termination of a task within a specified unit of work.
 - Replaces the Level 2 provided CALLRTM program which would accomplish the same thing, but required customization and installation
 - The task is terminated with a new code of ABEND80D-0
 - An address space can protect itself from the FORCE TCB command by turning on the ASCBNOFT bit.
- The command to terminate a specific task is:
 - FORCE *jobname*,TCB=*ttttt*[,RETRY=NO|YES][,A=asid]
 - **RETRY=** indicates whether the task's recovery processing is allowed to attempt retry.
 - IBM Service Level 2 guidance is required to identify the hung task

System Symbol for z/OS release

- New System Symbol for the z/OS release level
- **&SYSOSLVL** is automatically set to **Z1vvrrmm**, where:
 - **Z1** indicates z/OS
 - *vv* is the version, derived from field ECVTPVER
 - *rr* is the release, derived from field ECVTPREL
 - *mm* is the modification level derived from field ECVTPMOD
- For example, the value for z/OS 2.1 is **Z1020100**

Reserve SVC entry and still get Fxx Abend

- An installation can now define an SVC Table entry that does not actually have an associate SVC routine.
- The idea is to "reserve" an SVC of particular type and attributes so that it can be "activated" later.
- This is done in IEASVCxx by using the name IGCERROR which indicates to z/OS that even though it is "defined" it still should get the abend Fxx upon use.
- Since some programs find unused SVC Table entries by looking for ones that have an SVC routine address pointing to the IGCERROR module, entries reserved in this new way will not be pointed to the actual IGCERROR module.
- Entries reserved in this way look in-use but are known to be merely reserved by the operating system.

Tracing DIEs in System Trace

- A System Trace entry will be created upon exit from a timer Disabled Interrupt Exit (DIE).
- Tracing the DIE entry is not done since it can provide no additional information beyond that provided in the Interruption entry and the new DIE Exit entry.
- Data traced at DIE exit includes
 - Exit address
 - CPU time spent in the exit

Support for Time in the 2nd Epoch

- The CONVTOD and STCKCONV services will now support time values up to the end of the 2nd epoch.
 - The 1st epoch ends in September 2042.
 - The 2st epoch ends June, 2185 - or a STCKE value with less than x'02' in the first byte.
- Value beyond the end of the 2nd epoch are still rejected

Some New Health Checker Routines

- Check whether XTIOs (XTIOT = a TIOT (Task I/O table) above 16MB) are enabled for non-VSAM data sets.
- Checks if System Trace branch tracing has been active for a longer time than a given duration. Default is 4 hours.
- Checks if System Trace mode tracing has been active for a longer time than a given duration. Default is 4 hours.
- Checks if a PER trap has been active for a longer time than a given duration. Default is 4 hours.
- Checks whether VLF is trimming recently added objects while making room for new objects. Compares to AlertAge
- Check whether the GRS RNLs match IBM recommendations for reserves for SYSIGGV2, SYSZVVDS and SYSVTOC
- Check availability of kernel supply of stack cellpool cells based on a percentage used. Default is 85%.

Dynamically Add and Delete MCS Consoles

- *Prior to z/OS 2.1, an IPL is required to define a new console and a sysplexwide IPL is required to delete a console.*
 - *excepting the use of the SAMPLIB utility IEARELCN.*
- z/OS 2.1 introduces 2 new commands
 - The commands will only run in console support mode of Distributed.
- SET CON=xx will read the specified CONSOLxx parmlib member and process the operational settings as well as create new consoles.
 - Also, some additional settings can now be changed by the command
 - DEFAULT statement: HOLDMODE and SYNCHDEST
 - HARDCOPY statement: HCFORMAT
 - INIT statement: CMDDELIM and the CTRACE default
- SETCON DELETE,CN=nnnnnnnnn which will delete the specified console from the sysplex.

System symbol updates for local system

- A SETLOAD command option is provided to build a new, complete static system symbol table (not a delta) from a Parmlib member without an IPL
 - The IEASYMUP tool is not formally supported and its updates are transient (i.e., will not remain across a re-IPL)
- SETLOAD *xx*, IEASYM
 - The *xx* identifies a LOAD*xx* member whose IEASYM statement identifies the IEASYM*xx* parmlib member(s) to be used for the symbol table build
- On a re-IPL, if the same LOAD*xx* member is used, it will produce the same set of symbols.
- **Note:** updates are not sent to remote systems, so JES and ARM processing on remote systems on behalf of this system may continue to use the symbol table that they received when this system IPLed.

Line Comment in BCP Parmlib Members

- A line with an asterisk in column 1 is now treated as a comment in these Parmlib members:
 - COMMNDxx
 - GTF parmlib member (identified by GTF PROC)
 - IEAABD00
 - IEACMD00
 - IEADMP00
 - IEADMR00
 - IEAPAKxx
 - IEASYSxx
 - LPALSTxx
 - VATLST00