

z/OS 2.1 JES2 Product Update and Latest Status

Updated March 11th 2014



Tom Wasik
IBM Rochester, MN
wasik@us.ibm.com
Tuesday 9:30AM
Session Number 14661



Mainframe 50 April 7th 1964 - April 7th 2014
<http://www.ibm.com/mainframe50/>

Trademarks

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

- IBM®
- MVST™
- Redbooks®
- RETAIN®
- z/OS®
- zSeries®

The following are trademarks or registered trademarks of other companies.

- Java and all Java-related trademarks and logos are trademarks of Sun Microsystems, Inc., in the United States and other countries.
- Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.
- Microsoft, Windows and Windows NT are registered trademarks of Microsoft Corporation.
- UNIX is a registered trademark of The Open Group in the United States and other countries.
- SET and Secure Electronic Transaction are trademarks owned by SET Secure Electronic Transaction LLC.
- All other products may be trademarks or registered trademarks of their respective companies.

Notes:

Performance is in Internal Throughput Rate (ITR) ratio based on measurements and projections using standard IBM benchmarks in a controlled environment. The actual throughput that any user will experience will vary depending upon considerations such as the amount of multiprogramming in the user's job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve throughput improvements equivalent to the performance ratios stated here.

IBM hardware products are manufactured from new parts, or new and serviceable used parts. Regardless, our warranty terms apply.

All customer examples cited or described in this presentation are presented as illustrations of the manner in which some customers have used IBM products and the results they may have achieved. Actual environmental costs and performance characteristics will vary depending on individual customer configurations and conditions.

This publication was produced in the United States. IBM may not offer the products, services or features discussed in this document in other countries, and the information may be subject to change without notice. Consult your local IBM Business contact for information on the product or services available in your area.

All statements regarding IBM's future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only.

Information about non-IBM products is obtained from the manufacturers of those products or their published announcements. IBM has not tested those products and cannot confirm the performance, compatibility, or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

Prices subject to change without notice. Contact your IBM representative or Business Partner for the most current pricing in your geography.

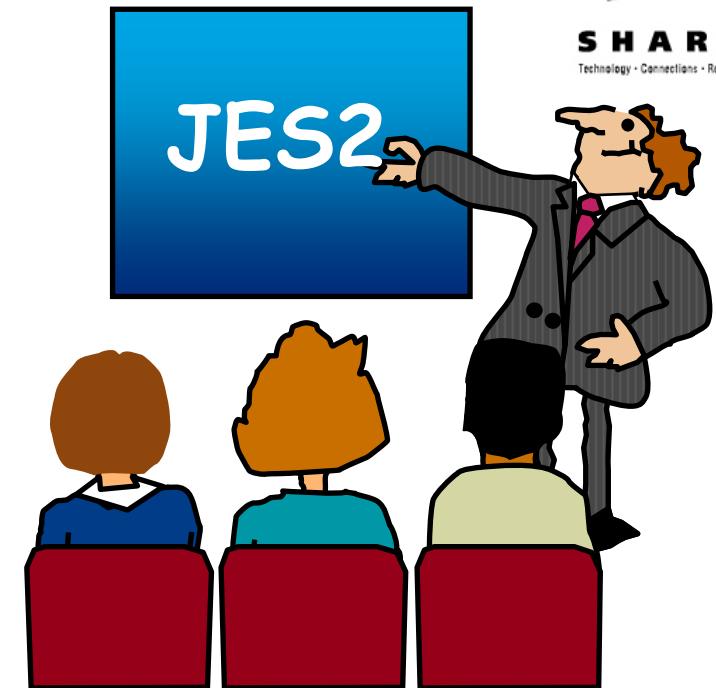


Agenda . . .



■ JES2

- ▶ Current JES2 Releases
- ▶ Migration & Implementation Tips
- ▶ Recent Service



▶ Other Presentation

- ▶ **JES2 BootCamp, Part 1 of 2 – What is JES2 and what does it do**
 - ▶ Wednesday 9:30 AM-10:30 AM – Platinum Ballroom Salon 8
- ▶ **JES2 BootCamp, Part 2 of 2 – Setup, Operations, and Customization**
 - ▶ Wednesday 11:00 AM-12:15 PM – Platinum Ballroom Salon 8
- ▶ **Batch Modernization in z/OS 2.1 JES2**
 - ▶ Thursday 8:00 AM-9:00AM – Grand Ballroom Salon C



Current JES2 Releases



FMIDs, Birthdays & Obituaries

| JES2 Rel.# | FMID | First Available | No Longer Available | End of Service |
|------------|----------------|-----------------|---------------------|----------------|
| z/OS 1.5 | HJE7708 | 3/04 | 9/04 | 9/07 |
| z/OS 1.6 | HJE7708 | 9/04 | 10/05 | 9/07 |
| z/OS 1.7 | HJE7720 | 9/05 | 10/06 | 9/08 |
| z/OS 1.8 | HJE7730 | 9/06 | 10/07 | 9/09 |
| z/OS 1.9 | HJE7740 | 9/07 | 10/08 | 9/10 |
| z/OS 1.10 | HJE7750 | 9/08 | 10/09 | 9/11 |
| z/OS 1.11 | HJE7760 | 9/09 | 10/10 | 9/12 |
| z/OS 1.12 | HJE7770 | 9/10 | 10/11 | 9/14* |
| z/OS 1.13 | HJE7780 | 9/11 | 01/14 | 9/16* |
| z/OS 2.1 | HJE7790 | 9/13 | | 9/18* |

See http://www-03.ibm.com/systems/z/os/zos/support/zos_eos_dates.html

*** = projected...**



JES2/MVS Compatibility



JES2 Release:

| z/OS Release | JES2 z/OS R.8 HJE7730 | JES2 z/OS R.9 HJE7740 | JES2 z/OS R.10 HJE7750 | JES2 z/OS R.11 HJE7760 | JES2 z/OS R.12 HJE7770 | JES2 z/OS R.13 HJE7780 | JES2 z/OS R2.1 HJE7790 |
|--------------|--------------------------|--------------------------|---------------------------|---------------------------|---------------------------|---------------------------|---------------------------|
| z/OS R8 | X | | | | | | |
| z/OS R9 | X | X | | | | | |
| z/OS R10 | X | X | X | | | | |
| z/OS R11 | | X | X | X | | | |
| z/OS R12 | | | X | X | X | | |
| z/OS R13 | | | | X | X | X | |
| z/OS R2.1 | | | | | | | X |

IBM recommends the same level of z/OS & JES2 throughout your plex.

-JES levels must match z/OS level starting in z/OS 2.1,

-See "**z/OS V1R13.0 Planning for Installation**" Ch. 4 (GA22-7504)

at http://publibz.boulder.ibm.com/cgi-bin/bookmgr_OS390/BOOKS/e0z2b1c0/4.5.1



JES2/MVS Compatibility

Statement of Direction

- z/OS 1.13 will be the last z/OS release that supports running a down level JES.
- As of z/OS 2.1, IBM only supports running the 2.1 JES on the 2.1 MVS
 - z/OS 1.13 and earlier will not be **supported** running on a z/OS 2.1 z/OS
- This does NOT affect what members can co-exist in a MAS
 - Just the mixing and matching of JES and MVS levels.
- New messages during initialization:
 - WTO - JES2 *level (HJExxxx)* is not supported running on *mvs_level*
 - WTOR - Reply "CONTINUE" to initialize JES2 in this unsupported environment, "TERM" to shut down JES2



z2 Checkpoint Mode

Statement of Direction

- z/OS 2.1 is planned to be the last release to support z2 checkpoint mode
 - \$ACTIVATE LEVEL=Z2
- z11 checkpoint mode was introduced in z/OS 1.11
 - \$ACTIVATE LEVEL=Z11
- Migrate to z11 mode if you have not already done so
 - See z/OS 1.11 presentation for information on impacts of z11 mode



Survey Questions



What is your JES2 Release level (are you \$ACTIVATED) ?

| JES2 Rel.# | SHARE 122 3/2014 | SHARE 121 8/2013 | SHARE 120 2/2013 | SHARE 119 8/2012 | SHARE 118 3/2012 | SHARE 117 8/2011 | SHARE 116 2/2011 | SHARE 115 8/2010 | SHARE 114 3/2010 | SHARE 113 8/2009 | SHARE 112 3/2009 | SHARE 111 8/2008 | SHARE 110 2/2008 | SHARE 109 8/2007 |
|---------------|------------------------|------------------------|------------------------|------------------------|------------------------|------------------------|------------------------|------------------------|------------------------|------------------------|------------------------|------------------------|------------------------|------------------------|
| z/OS R4 | | | | | | | 1 | | | | | | 1 | 4(4) |
| z/OS R5 | | | | 1 | | | | 1 | | | | | | 6(6) |
| z/OS R7 | | | | 1 | | | | | | 2 | 1 | 2 | 12 | 19 |
| z/OS R8 | | | | 1 | | | 1 | | | 3 | 4 | 14 | 13 | 18 |
| z/OS R9 | | | | 2 | 1 | 1 | 5 | 5 | 19 | 31 | 14 | 16 | 5 | |
| z/OS R10 | | 1 | | 1 | | 3 | 8 | 9 | 12 | 20 | 5 | | | |
| z/OS R11 | 1(1) | | 4(3) | 6(5) | 21(19) | 20(9) | 26(7) | 12(5) | 9(2) | 3(0) | | | | |
| z/OS R12 | 7(5) | 9(8) | 7(4) | 10(10) | 15(15) | 14(7) | 13 | | | | | | | |
| z/OS R13 | 24(23) | 32(30) | 10(8) | 11(11) | 8(8) | | | | | | | | | |
| z/OS R2.1 | 5(5) | | | | | | | | | | | | | |



z/OS 2.1 Overview

- Batch Modernization
 - New JCL keywords
 - Limit jobs by MVS level
 - 8 Character job class
 - SAF checks for JOBCLASS access
 - Calling interpreter after converter
 - System symbols in batch jobs
 - Exporting symbols to runtime
 - New symbols services
 - Symbolic substitution in instream data sets
 - Symbols on INTRDR
 - Job Correlator support
 - Job Completion ENF 78
- Other enhancements
 - Job Modify SSI
 - Performance Updates
 - Miscellaneous changes



New JOB JCL keywords

- **SYSAFF=** – List of JES2 members where job can run
 - Same as JECL JOBPARM SYSAFF=
- **SYSTEMS=** – List of MVS system names where job can run
 - Similar to SYSAFF but a list of MVS system names
 - Does not support independent more (IND)
- **UJOBCORR=** – Specifies user portion of job correlator
 - Can also be specified on the Start command
 - S DEALLOC,UJOBCORR=TEST



New OUTPUT JCL Keywords

- **MERGE=YES** – Merge operands on all SYSOUT
 - Single default for JCL keywords
 - Provides base default for the job
 - Does not create new instance of a data set
 - Applies even if other OUTPUT card referenced
 - Replaces functions provided by various JECL statements
- **DDNAME=** – Forward pointing OUTPUT specification
 - Similar to DDNAME on JES3 *//*FORMAT* card
 - DDNAME= *ddname* or *step.ddname* or *proc.step.ddname*
 - Allows easy specification of OUTPUT cards to DD statement
 - One instance per matching OUTPUT card
 - If DDNAME matches, then DEFAULT=YES does not apply
 - Can be used with DD OUTPUT= specified OUTPUT cards (creates more instances)
 - If DDNAME= and DD OUTPUT= matches same OUTPUT card, creates 1 instance



New JCLLIB JCL Keyword

- **PROCLIB=** – Specifies the DDNAME of the PROCLIB concatenation to use
 - Same as JECL JOBPARM PROCLIB= statement
 - Preferred to using JECL
- JCLLIB PROCLIB= overrides JOBPARM PROCLIB=
- Static PROCLIBs displayed with \$D PROCLIB
 - Can be “altered” by command (creates new concatenation) JES2 PROCLIB support has been updated
 - Display now includes concatenation in the JES2 PROC
 - Displayed as “STATIC PROCLIB”s
 - Can be override as in previous releases
 - Static PROCLIBs can be “altered” more easily using \$T PROCLIB
 - Creates override based on static values
 - Quicker close/unallocate of unused concatenations



Eight Character Job Class

- JES2 supports up to 8 character job class names
 - Similar to existing support in JES3
- JCL JOB card CLASS= will be expanded to support up to 8 characters
- Classes will be managed by \$ADD/\$DEL JOBCLASS command
- Other commands updated to support 8 character job classes
- Any job classes can be set inactive (ACTIVE=NO)
 - Stops NEW jobs from specifying the job class
 - Does not affect existing jobs or job selection
 - Causes same error as undefined class
 - Applies to single and multi character classes
- New default JOBCLASS for use in error cases
 - JOBDEF DEF_CLASS= specifies value



Job Class Groups

- Each job class can be in one group
- Group name and job classes must be unique
 - Cannot have a job class group with the same name as a job class
- Facilitates selecting on job classes
 - Inits and Offload Job Transmitters can specify either
 - 1-36 single character job classes
 - 1-8 multi (or single) character job classes or job class groups
- Selection from groups is done round robin
 - When a job is selected, the classes are rotated so the next select will start with the next job class in a group
- CLASS= parameter updated (command & init statement)
 - CLASS=ABCD – implies single character job classes A, B, C, D
 - CLASS=(ABCD) – implies 4 character job class or job class group ABCD



JOBCLASS commands

- The \$ADD JOBCLASS command defines a new job class
 - Same operands as \$T JOBCLASS or JOBCLASS init statement
 - Class name cannot match an existing job class group
 - All members must be z/OS 2.1
 - Must be in z11 \$ACTIVATE mode

```
$ADD JOBCLASS ( CLASS1 ) , ACTIVE=YES , MODE=WLM
```

```
$HASP837 JOBCLASS ( CLASS1 )          ACTIVE=YES , GROUP= , MODE=WLM ,
$HASP837                               QAFF= ( ANY ) , QHELD=NO , SCHENV= ,
$HASP837
      XEQCOUNT= ( MAXIMUM=* , CURRENT=0 ) ,
$HASP837                               XEQMEMBER ( IBM1 ) = ( MAXIMUM=* ,
$HASP837                               CURRENT=0 )
```



JOBCLASS commands

- The \$DEL JOBCLASS command deletes a job class
 - Cannot delete 1 character job classes or STC/TSU
 - Class must be set to ACTIVE=NO before deleting it
 - Class deleted from
 - Group (if any) it is in
 - Any selection list (removed from list)
 - Default job class (replace with JOBDEF default class)
 - *Cannot delete from all default setting – be careful*
 - Filters are supported

```
$DEL JOBCLASS (CLASS1 )
```

```
$HASP837 JOBCLASS (CLASS1 )           - ELEMENT DELETED
```



JOBCLASS commands

- \$T JOBCLASS command updated with new operands
 - Is job class usable for new jobs entering the systems
 - ACTIVE=YES|NO
 - Specifies the job class group the job is in
 - GROUP=*grpname*

```

$T JOBCLASS ( CLASS1 , CLASS2 ) , GROUP=XYZ , ACTIVE=YES
$HASP837 JOBCLASS ( CLASS1 ) ACTIVE=YES , GROUP=XYZ , MODE=JES ,
$HASP837 QAFF= ( ANY ) , QHELD=NO , SCHENV= ,
$HASP837
    XEQCOUNT= ( MAXIMUM=* , CURRENT=0 ) ,
$HASP837 XEQMEMBER ( IBM1 ) = ( MAXIMUM=* ,
$HASP837 CURRENT=0 )
$HASP837 JOBCLASS ( CLASS2 ) ACTIVE=YES , GROUP=XYZ , MODE=JES ,
$HASP837 QAFF= ( ANY ) , QHELD=NO , SCHENV= ,
$HASP837
    XEQCOUNT= ( MAXIMUM=* , CURRENT=0 ) ,
$HASP837 XEQMEMBER ( IBM1 ) = ( MAXIMUM=* ,
$HASP837 CURRENT=0 )

```



CLASSGRP commands

- New command to display job class groups
 - Displays group name and list of classes in the group
 - `$D CLASSGRP(name)`

```
$D CLASSGRP
```

```
$HASP816 CLASSGRP (TEST)
```

```
B , NONE , A
```

```
$HASP816 CLASSGRP (XYZ)
```

```
CLASS1 , CLASS2
```

Specifying job classes

- Specifying job class selection list (CLASS=)
 - Applies to \$T INITnn, OFFn.JR and OFFn.JT
 - Two syntax now supported
 - List of 1-36 single character classes
 - List of 1-8 job classes or job class groups
- Traditional syntax - 1-36 single character classes
CLASS=**ABCD**
- New syntax – 1-8 classes or class groups
CLASS= (**CLASSX , CLASSY , GROUP1**)
CLASS= (**A , B , C , D**)
- Presence of parenthesis indicates syntax in use
 - Beware of CLASS=TEMP vs CLASS=(TEMP)
 - First is 4 classes T,E,M,P second is 1 class (or group) TEMP



Initialization statements

- New JOBCLASS operands added to initialization statement
 - ACTIVE and GROUP operands
 - As with other operands, only applies on cold start
- New job classes can be added on any start
 - Any JOBCLASS in the init deck that does not exist is added
 - All members must be 2.1 and must be z11 \$ACTIVATE mode
 - Operands set (or defaulted) on new class are honored
 - CAN RESURRECT DELETED JOB CLASS
 - *Always keep init deck current*
- Cannot delete JOBCLASS using init deck
- Changes to CLASS= specification also apply to init statements
 - CLASS=TEMP vs CLASS=(TEMP)



JCL, SMF records, etc.

- CLASS= on JOB card updated to support 1-8 characters
- General philosophy for 1 character class fields in multiple CBs
 - Existing 1 byte fields set to 1st character of multi character class
- MVS JMR updated to support longer class value
 - New extended JMR created to add space to the JMR
 - At end of current JMR area
 - JMRVERSN set to 1 if extension exists
 - New job class field JMRCLAS8 (always set to job class)
- SMF field SMF26WJC set to full job class value
- SMF field SMF30CL8 (formerly JES3 only) used for JES2 class value
- SSIs already had 8 character fields for job class
 - Code now populates/uses entire 8 character field



Coexistence

- MAS with down level releases supported (with appropriate maintenance)
 - However cannot start down level member if
 - A multi character job class is defined
 - A job is in a job class group
 - If a job class is ACTIVE=NO
 - Down levels can start but they ignore ACTIVE setting
- Cannot \$ACTIVATE to z2 level if
 - A multi character job class is defined
 - A job is in a job class group



Exit considerations (exploitation actions)

- Once 8 character job classes defined:
 - 36 class queue heads in the \$HCT (\$JQHEAD1) not updated
 - For new and existing classes
 - JQETYPE does not reflect job class for 8 character classes
 - JQETYPE unchanged for 1 character classes
 - 8 character field set in new field in \$JCT (in an IBM extension)
- To simplify exit writing do not update class in JCT/JQE if possible
 - Use fields in XPLs to update job class (1 or 8 byte value)
 - Exit 6 updated with XPL and field in XPL to set class
 - Other exits (2, 4, 50, 52, 54 etc) already had class fields



SAF/RACF checks for job classes

- JES2 now supports verifying a jobs access to a job class
 - This eliminates a number of customer exits to do similar check
- The check verifies READ access to a entity name in the JESJOBS class
 - `JOBCLASS.nodename.jobclass.jobname`
 - *nodename* is the local NJE node name
 - *jobclass* is the job class to be associated with the job
 - *jobname* is the job name of the job being submitted
- The check is activated by 2 profiles in the FACILITY class
 - JES.JOBCLASS.OWNER controls an owner check
 - JES.JOBCLASS.SUBMITTER controls a submitter check
 - If both profiles exist, 2 AUTH checks are made
 - *Both checks must PASS for job to continue*
 - Only search for discrete (non generic) profiled in the FACILITY class
 - Access list and UACC of FACILITY class profiles are **not** used



SAF/RACF usage example

- User TOM submits a CLASS=C job named TEST with USER=TONI
 - NJE Note name is POK
- If there is a JES.JOBCLASS.OWNER profile in the FACILITY class
 - A check is made if user TONI has READ access to JESJOBS profile
JOBCLASS.POK.C.TEST
- If there is a JES.JOBCLASS.SUBMITTER profile in the FACILITY class
 - A check is made if user TOM has READ access to JESJOBS profile
JOBCLASS.POK.C.TEST
- If both FACILITY class profiles exist, then TOM and TONI must have access to the JESJOBS class profile



JESJOBS class considerations

- Activating JOBCLASS checking considerations:
 - Ensure profiles exist to permit jobs that need to run have access
 - Start with a JOBCLASS.** profile with UACC of read
 - *Blanket permits every job to run*
 - Add profiles to restrict job classes
 - *Consider warn mode at first*
 - *JOBCLASS.*.R.* restricts class R for all job names and nodes*
 - If JESJOBS class never used
 - Ensure before starting that SUBMIT.** has appropriate access
 - Review documentation on CANCEL.**
 - Refrain from a * profile with UACC other than NONE
 - If JESJOBS class is in use, check for * profiles
 - OK if UACC of NONE, not a good idea if not NONE
- Create facility class profile AFTER setting up JESJOBS profiles.



Interpreter After Converter

- Allows processing of OUTPUT JESDS= if job not executed
- More complete checking with TYPRUN=SCAN
- New option to call interpreter after conversion
 - JOBDEF INTERPRET=JES|INIT
 - Must be z11 mode and all members z/OS 2.1 before option takes effect
 - If down level joins MAS, revert to prior release processing

Converter/interpreter run in a separate address space

- No change if not calling interpreter
- CISUB_PER_AS= controls number of task per C/I address space
- Migration impact to exit 7
 - Exit 8 not 7 always called to read/write IOTs for conversion
- EXIT 6 only called in JES2 address space
 - New exit 60 called in JES2CI address space (same function as exit 6)
 - No access to HCT or JES2 private in exit 60



Performing interpretation after conversion

- If INTERPRET=JES is selected
 - Both converter and interpreter run in a new address space
 - Address space name *jesxClxx*
jesx subsystem name, *xx* is an instance number
 - Define address space to RACF same as JES2
 - *Needs access to all PROCLIB data sets*
 - *Does VERIFY CREATE for jobs being converted*
 - PCEDEF CNVNUM= controls number of converters run
 - CISUB_PER_AS=*nn* – indicates how many subtasks per JES2CI
 - Default is 5 per address space
 - Default number of converters is 10
 - Default number of JES2CI address spaces is 2 (10/5)
JES2CI01 and JES2CI02



Performing interpretation after conversion

- Implications of code change
 - CNVT code to read IOT CBs moved to subtask
 - Calls CBIO exit 8 instead of exit 7
 - Exit 6 compatibly updated with this release
 - Same data passed to exit in registers 0 and 1
 - New is an XPL in register 2
 - *XPL makes it easier to set job CLASS, SCHENV, and HOLD*
 - Existing exits should continue to function without change
 - *Presuming they use data areas passed to the exit*
 - *CNVT DTE (\$DTECNV) structure has changed significantly*
 - *Many field moved to \$CIWORK*
 - New internal data set to store interpreter output
 - \$SWABLKS DD name
 - Always created by z/OS 2.1 JES2 during input phase



Performing interpretation after conversion

- Implications of setting INTERPRET=JES
 - Converter and interpreter run outside the JES2 address space
 - Exit 8 cannot access JES2 private storage
 - New exit 60 to perform function of exit 6
 - Exact same parameters as exit 6 (New XPL in register 2)
 - *XPL includes setting job CLASS, SCHENV, and HOLD*
 - Gets control in USER environment (R11 is HCCT)
 - Run in JES2CI address space (cannot access JES2 private storage)
 - New sample exits HASX60A and HASX60B
- New exit 59
 - Invoked after interpreter call and after processing OUTPUT cards
 - Invoked before writing data to \$SWABLKS DD
 - Same XPL as exit 60 (but passed in register 1)



Converter Issues Input Error/Warning Messages

- Input phase detected errors currently written to JCLIN
 - Causes confusion for input detected errors
 - JCLIN data set becomes only print data set
- Input errors will now be passed to Converter
 - JECL assigned statement numbers
 - Added to standard message data set with other errors
 - Job will not be failed at input time but queued to converter
 - Will be forced to convert on the input member
 - Interpreter called to ensure OUTPUT JESDS= will be processed
- Consistent processing of JCL errors
 - JES2, Converter, and Interpreter error processed the same



System Symbol Substitution in BATCH Jobs

- Any system symbol (IEASYMxx) can be referred to in JCL
 - JOBLCLASS SYSSYM=ALLOW|DISALLOW option
- Substitution occurs at conversion time
- Control conversion using scheduling environment
 - JOBDEF CNVT_SCHENV=IGNORE|HONOR
- Provides consistent conversion and execution symbols
- Can also use traditional affinity
 - JOB JCL and JOBPARM JECL



Exporting Symbols to Run Time Environment

- Makes JCL symbols available to running job
- **EXPORT SYMLIST=(xxx,yyy....)** JCL identifies symbols
- Subsequent SET statements are exported to run time
 - Only need to identify a symbol to export once
- Only one value of symbol per step
 - Last SET statement value is exported



New Symbol Services

- JCL symbol service IEFSJSYM
 - All exported symbols with value for step can be accessed
 - Access is READ ONLY, cannot be updated or deleted
- JES symbol service IAZSYMBL
 - Provides dynamic symbols at the step or TCB level
 - Symbols can be created/deleted/updated by runtime
 - Application variable symbols
 - Special system use symbols for passing data
 - Can access JCL symbols using this service
 - Symbol names can be 16 characters
 - Symbol values can be up to 4K



Symbolic Substitution in Instream Data Sets

- **SYMBOLS=** keyword on DD DATA or DD * JCL
SYMBOLS=[() JCLONLY | EXECSYS | CNVTSYS [, *DDname*])
 - JCLONLY – Substitute **EXPORT**ed JCL symbols and current JES symbols
 - EXECSYS – JCL plus system symbols from execution system
 - CNVTSYS – JCL plus system symbols from conversion system
Consistent with system symbol substitution
 - *DDname* – DD name for LOG data set
 - Not specified – No substitution (default and current processing)
- Valid for JES and Converter created instream data sets
- Substitution as records are read by application
- Blank elimination to try to make things fit
 - Overflow if possible, I/O error if not
- Log will track what was passed to application
 - Original and substituted text available
 - DD name for LOG data set passed in
 - Does not work with NEW PARMDD= on EXEC card



Passing Symbols on INTRDR

- JES or JCL symbols can be passed on INTRDR
- Become initial SET symbols for jobs submitted
 - Automatically exported
- **SYMLIST=** keyword identifies symbols to pass
 - Only symbols that obey JCL rules can be passed
 - Generics can be used to identify symbols (eg * or DSN*)
- Values extracted when JOB card is processed
- Values can be updated between jobs



New Job Correlator Function

- 64 byte unique identifier to track and manage jobs
 - 32 characters of system information
 - 32 characters of application data
 - “:” delimiter between 2 sections
 - Last character in system section
- Specify application section using reserved JES symbol
 - SYS_CORR_USRDATA – 32 characters
 - Set before submitting job card
- Last submitted jobs correlator set in task level JES symbol
 - SYS_CORR_LASTJOB – 64 characters
- Correlator of current job set in step level JES symbol
 - SYS_CORR_CURRJOB – 64 characters
- JOBID of last job submitted at task level JES symbol
 - SYS_LASTJOBID – 8 characters



Using the Job Correlator

- Correlator can be used (with wildcards) for selection on
 - Extended status SSI
 - SAPI SSI
 - JES2 job list command
- Returned on request job ID SSI
- Passed on ENF 58 (SYSOUT) and 70 (JOB)
- Address space correlator available using IAZXJSAB
 - From any address space
- Added to JMR
- Added to SMF records



New ENF 78 for Job Completion

- Multi system ENF issued when job is past execution
- Only issued if there is a notify value set for job
 - Set before job card is submitted
 - JES symbol SYS_JOB_NOTIFY - up to 4K bytes
- ENF data is similar to ENF 70
 - Job information, correlator
 - Job completion information
 - SYS_JOB_NOTIFY value



Job Modify SSI service

- New SSI to modify jobs
 - Unauthorized SSI to modify jobs and their characteristics
 - Input/Filters similar to Extended Status SSI
 - JESJOBS SAF checks to authorize user can affect changes
 - Requests can be synchronous or asynchronous
- Functions supported
 - Modify job characteristics (\$T) – Hold a job (\$H)
 - Release a job (\$A)
 - Purge a job (\$P)
 - Cancel a job (\$C) – with options to purge and/or dump
 - Restart a job (\$E) – with the cancel or step and hold options
 - SPIN a job (\$T, SPIN) – with the optional DDNAME option
 - Change execution node (\$R XEQ) – only if pre-execution
 - Start a job (\$S)



Job Modify SSI Services SAF entities

- SAF entities checked for other requests
 - Requesting address space userid checked against entity names

| SSI Action | JESJOBS Class Entity | Access |
|-------------------|---------------------------------------------------|---------|
| Modify | MODIFY.nodename.userid.jobname | Update |
| Hold | HOLD.nodename.userid.jobname | Update |
| Release | RELEASE.nodename.userid.jobname | Update |
| Purge | PURGE.nodename.userid.jobname | Alter |
| Cancel | CANCEL.nodename.userid.jobname - Existing profile | Alter |
| Restart | RESTART.nodename.userid.jobname | Control |
| Spin | SPIN.nodename.userid.jobname | Control |
| Reroute execution | REROUTE.nodename.userid.jobname | Update |
| Start | START.nodename.userid.jobname | Control |



SAPI performance improvements

- New tree structures to improve SAPI performance
 - Improves search for jobs for SAPI selection
 - Improves SAPI search when new output is created
- Controlled by parameters on OUTDEF
 - SAPI_OPT=YES|NO controls SAPI notification (\$#POST)
 - WS_OPT=YES|NO controls SAPI selection performance (\$#GET)
 - Requires all members z/OS 2.1 and \$ACTIVATE LEVEL=z11
- Benefit requires selection on at least
 - QUEUE,ROUTECD,OUTDISP
 - QUEUE,OUTDISP
 - ROUTECD,OUTDISP
 - OUTDISP
- Consider using optimized selection criteria
 - Selecting on a single criteria such as writer will not be optimized



SAPI performance improvements

- New JOE type (index JOE) implements one of the trees
 - Consumes additional JOEs based on number of unique combinations of
 - SYSOUT class, destination, and OUTDISP
- Index JOEs are never “cleaned up” in this release
 - Every unique combination of characteristics creates an index JOE
 - Once index JOE is created, it persists until index is rebuilt
 - Index is rebuilt when it is toggled off and on by setting SAPI_OPT
 - New command \$D OUTDEF,JOEUSE displays how JOEs are used

```
$doutdef , joeuse
$HASP836 OUTDEF
$HASP836 OUTDEF CURRENT JOE UTILIZATION
$HASP836 TYPE COUNT
$HASP836 -----
$HASP836 WORK 104
$HASP836 CHAR 1
$HASP836 INDEX 3
$HASP836 FREE 92
```



SAPI performance improvements – POST expected results

- Results from a measurement in the following scenario:
 - 50 SAPI printers selecting work by unique destination, waiting for output to become available
 - One job runs and creates one output group
 - Measurement is done in a steady state
- CPU cost of SAPI POST with SAPI_OPT=YES is 3.5 times smaller than with SAPI_OPT=NO when no device is selected (output does not match any device)
- CPU cost of SAPI POST with SAPI_OPT=YES is 1.8 times smaller than with SAPI_OPT=NO when a device is selected
- CPU cost of processing with SAPI_OPT=NO and SAPI_OPT=YES scale differently, so with larger number of devices larger performance benefit is expected



SAPI performance improvements – GET expected results

- Optimized path for active SYSOUT work selection (GET) uses index structure over JOEs
- CPU cost of the optimized path depends on many factors:
 - Number of selectable JOEs
 - A match between selection requests and the JOE index.
 - If selection request does not match index, the request goes through the old path
 - Statistical distribution of JOE attributes
 - e.g. many unique destinations in multiple classes vs many JOEs with the same class and destination
- This makes it hard to predict expected performance in all possible environments



SAPI performance improvements – GET expected results

- Results from a measurement in the following scenario:
 - 15000 JOEs selectable JOEs
 - 3000 JOEs match the selection expression
- CPU cost of selecting a JOE with WS_OPT=YES is about 17.6 times smaller than with WS_OPT=NO when a JOE is selected
- Processing with WS_OPT=NO and WS_OPT=YES scale differently, so with larger number of JOES larger performance benefit is expected



SAPI performance improvements – GET cautionary statement

- Optimized path for active SYSOUT work selection (GET) uses index structure over JOEs
- This JOE index requires maintenance, which has its own CPU overhead.
- It is unlikely but possible that in environment where majority of job SYSOUT is never processed (never selected by SAPI applications), the overhead of index maintenance will not be offset by the performance improvements of work selection. In this case, using old path (WS_OPT=NO) may be preferable.
- Note index JOEs are never deleted
 - May need to clean up JOEs by toggling WS_OPT no and yes



Miscellaneous Changes

- Internal readers can be allocated from any address space
 - No need to get a JES environment to submit a job
- \$D INITINFO - display initialization information command
 - Start command, Init decks used, STEPLIB concatenation
- JOBCCLASS DSENQSHR=AUTO|DISALLOW|ALLOW
 - Supports ENQ downgrade changes
- Extended status output can be in 64 bit storage
- Extended status data set list can suppress duplicates
- SAPI returns data set number of returned data set



z/OS 2.1 Migration/Coexistence

- APAR OA36155 is needed for coexistence with 2.1
 - From JES2 z/OS 1.10 or z/OS 1.11 (with extended support)
 - From JES2 z/OS 1.12 or z/OS 1.13
- APAR also highly recommended for fall back as well
 - Jobs created on 2.1 may not be processed properly on earlier releases without OA36155
- **APARs OA43531 and OA43851/OA44178 also required**
- Exit 6 only runs in JES2 address space
 - New exit 60 with same function runs in user address space
 - Exit 60 must be in common storage
 - If running in JES2Clxx address space, then no HCT access
- Converter IOT read moved from exit 7 to exit 8
 - From JES2 to user environment
- \$HCT Queue heads not updated if multi-character jobclass exists
- Need to verify JESJOBS profiles for new job modify SSI



JES2 Service Information



HiPer APARs (Hi Impact, or Pervasive)

| <u>yy/mm/dd</u> | <u>APAR</u> | <u>COMMENTS</u> |
|-----------------|----------------|------------------------------------------------------------------------------------|
| 14/02/10 | OA44434 | MSGHASP107 When a JCLLIB statement is coded with ORDER= more than 256 characters |
| 13/12/09 | OA44065 | ABEND\$K03 RECEIVED WHEN CKPTDEF STATEMENT IS INCORRECTLY CODED |
| 13/11/14 | OA43851 | Additional 2.1 compatibility APAR - Non-standard job class prevents initialization |
| 13/10/18 | OA43658 | \$HASP096 DISASTROUS ERROR AT QCLASSER in 2.1 with NJH |
| 13/10/07 | OA43531 | Additional 2.1 compatibility APAR - jobs hung awaiting conversion |
| 13/09/17 | OA43367 | STC/TSU CLASSES HONOR BATCH CLASS KEYWORDS |
| 13/09/12 | OA43333 | SAPI APPLICATION FAILS TO PROCESS NEW SYSOUT |
| 13/08/19 | OA43123 | \$QXX ABENDS AFTER ISSUING A \$T CKPTDEF,MODE=DUPLEX |
| 13/07/12 | OA42787 | JOBS WITH SAME NAME RUNNING CONCURRENTLY DESPITE DUPL_JOB=DELAY |
| 13/06/04 | OA42452 | INCOMPLETE \$CDCT BUILT FOR JES2 DEVICES IN ERROR AT STARTUP |
| 13/03/26 | OA41793 | CORRUPTED FREE \$JOE CHAIN MAY CAUSES LOOP IN HASPJOS (\$HASP9202) |
| 13/03/26 | OA41768 | SAPI & PSO PCE DEADLOCK FOR JOBLOCK AND BERTLOCK RESOURCES |
| 13/02/27 | OA41571 | ABEND \$Q01 after issuing command with certain null operands |
| 13/02/07 | OA41400 | FSS PRINTER FAILS TO RESTART DUE TO OUTSTANDING ORDER CONDITION |
| 13/01/31 | OA41330 | Wild branch out of FRETRE routine when \$VERIFY is called |
| 12/11/26 | OA40901 | \$Q14 ABEND when SYSOUT transmitter updates unserialized JQE |
| 12/11/13 | OA40790 | \$HASB recovery code results in \$SJB chaining errors/loops needing IPL |
| 12/07/17 | OA39972 | ABEND 0C4 submitting jobs to an internal reader |
| 12/07/02 | OA39891 | SEGMENT= or spun data sets fail to print |
| 12/06/08 | OA39737 | After OA36256, JES2 fails to completely clean up a spool volume |
| 12/04/11 | OA39337 | AFTER OA37764, POSSIBLE MSGHASP401 WHEN STARTING JES2 |



Job and SYSOUT ENFs

- JES issues ENFs to notify JOB or SYSOUT events
 - Used by applications to track job or SYSOUT progress
 - ENF 58 – SYSOUT processing tracking
 - ENF 70 – JOB processing tracking
- Events creating ENFs include
 - Object create and delete
 - Selection and deselection
 - Phase or queue changes for jobs
 - Print progress via checkpoints for SYSOUT (upon request)
- Number of ENFs has increased in recent releases
 - Job level ENFs
 - High level ENFs for SYSOUT
 - SAPI application can request details for SYSOUT



Job and SYSOUT ENFs

- Intent of ENFs is to reduce “polling” for status
 - Push or event driven vs regular queries
 - Goal is to reduce overall system CPU needs
- Vendors are aware of capability
 - Not sure which have implemented listening to the ENFs
- ENFs are sent to all members of SYSPLEX (multi-system)
 - Cannot always predict where JOB/SYSOUT will be processed
 - Allows single application instance to monitor entire SYSPLEX
- Multiple JESPLEXes in SYSPLEX can increase total ENFs
 - Especially when jobs/SYSOUT sent to other JESPLEX for processing
 - ENFs for original instance and again for destination node
- May notice increased XCF traffic for ENF processing
 - Group associated with messaging is SYSENF



Performance APARs

- ENF performance enhancements
 - OA41698 – APAR limits ENFs to MAS vs SYSPLEX
 - MASDEF ENFScope=SYSPLEX|JESPLEX
 - Default is SYSPLEX
 - OA42099/OA41738 – Reduce CPU overhead of ENFs
 - Filters out JES2 listening to certain ENFs
 - Reduces overhead when ENF is received if no other listeners
 - OA41738 is conditioning APAR and must be applied first
- JES Property SSI, SPOOL information sub-function
 - Returns information including spool migration stats
 - Migration stats take significant CPU
 - OA43168 added switch to skip returning migration info
 - JPSP1MGR bit in IAZJPSPL



Performance APARs

- SAPI performance APAR - OA43882
 - Improves JES2 dispatcher priority scheme
 - Ensures CKPT process has proper priority
 - Corrected problem related to extending CKPT cycle based on activity on the system
 - Added code to prevent SAPI from monopolizing the system
 - Improved throughput when SAPI is processing SPIN data sets
- APAR diagnostics enhanced
 - New rolling CTRACE added to track SAPI requests
 - New/enhanced PERFDATA for \$#GET/\$#POST



PERFDATA updates (OA43882)

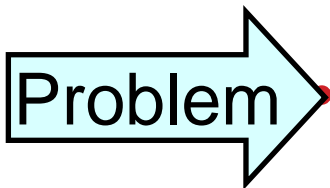
```

$Dperfddata(wsstat)
$HASP660 $DPERFDATA(WSSTAT)
$HASP660 WS PERFORMANCE STATISTICS IBM1-INTERVAL=1:15:07.944840
$HASP660 ---$#POST calls
$HASP660 CALL_COUNT=2006,AVERAGE_CPU=0.000009
$HASP660 ---Non-optimized $#GET calls
$HASP660 FOUND_COUNT=1000,AVERAGE_CPU=0.000020,
$HASP660 NONE_COUNT=2,AVERAGE_CPU=0.000001
$HASP660 ---Fast Path $#GET calls
$HASP660 FOUND_COUNT=1,AVERAGE_CPU=0.000029,
$HASP660 NONE_COUNT=1,AVERAGE_CPU=0.000000
$HASP660 ---Indexed $#GET calls | Index in
$HASP660 FOUND_COUNT=0,AVERAGE_CPU=0.000000, | z/OS 2.1
$HASP660 NONE_COUNT=0,AVERAGE_CPU=0.000000 | only

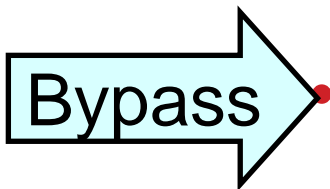
```



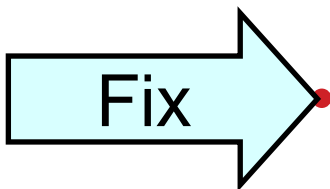
OA44511 – \$HASP896 RC83 for a volume in stunted at warm start



- Incorrect warm start validation of stunted volume
 - Buffers per track group wrong
- Stunted volume results when TGSPACE too small
 - Not all of volume is usable
 - Displayed in SDSF spool display



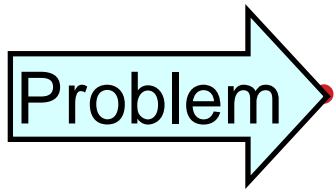
- Increase the TGSPACE=MAX= value
 - In general should not have stunted volume



JES2 z/OS 1.12, 1.13, 2.1

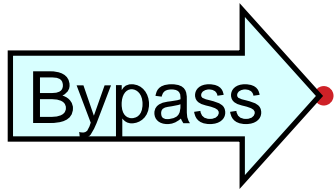


OA44434 – JCL error when JCLLIB coded with ORDER= more than 256 characters

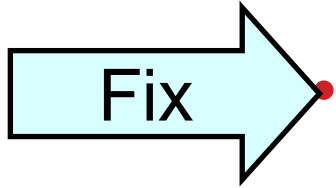


JCLLIB statement being flagged as in error by JES2

- ORDER= data set list greater than 256 characters



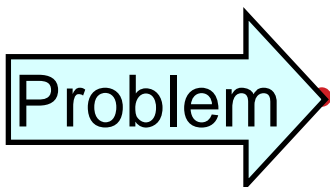
Reduce size of ORDER= keyword



JES2 z/OS 2.1

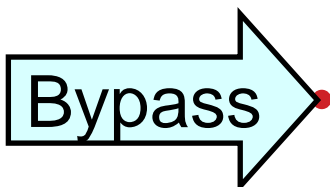


OA44065 – ABEND\$K03/ABEND0C4 with null CKPTDEF CKPT1/CKPT2



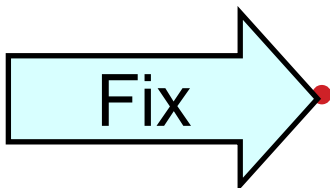
Problem

- A CKPTDEF statement is coded with either
 - DSN=/VOL= being null when CKPT is on CF
 - STR= being null when CKPT is on DASD
- Initialization messed up CKPT allocations
- JES2 cannot warm start (ABENDs 0C4)
- Other JESPLEX members ABEND K03
 - Must IPL, cannot hot start



Bypass

- Do not code null DSN=, VOL=, or STR=

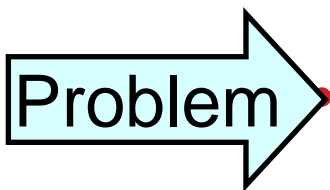


Fix

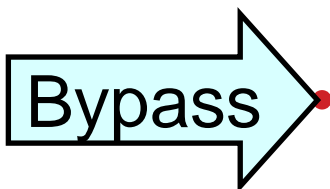
- JES2 z/OS 1.12, 1.13, and 2.1



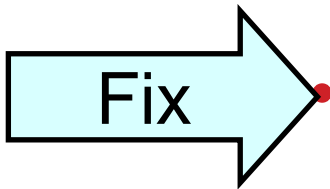
OA43851/OA44178 – Initialization failure due to non-standard job class



- A non-standard JOBCLASS was created prior to z/OS 2.1 (via an exit)
- z/OS 2.1 incorrectly assumed this was a \$ADDED JOBCLASS
- Once 2.1 started, older releases would not initialize
- Only problem if class is not A-Z, 0-9, STC or TSU



• None



• JES2 z/OS 1.12, 1.13, and 2.1



OA43708 – \$HASP443 RC03 WHEN SPOOL REALLOCATED

Problem

- Active SPOOL data set deleted by mistake
- Data set recreated in same place using ABSTR
- JES2 will not use data set, get
\$HASP443 <spool> DATASET SYS1.HASPACE NOT
ALLOCATED I/O ERROR DURING READ
VALIDATION
- JES2 made more tolerant of situation

Bypass

- Don't delete active spool volumes

Fix

- JES2 z/OS 1.12 and 1.13 (in 2.1 base)



OA43367 – STC/TSU CLASSES HONOR BATCH CLASS KEYWORDS

Problem

- JOBCLASS(*) init statement can include STC/TSU
- Keywords such as ACCT, HOLD, PGMNAME honored for STC/TSO
- Start commands can take JCL error for missing programmer name, be held, etc

Bypass

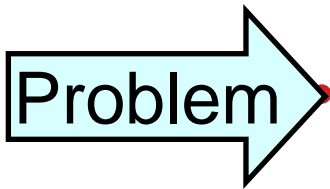
- Generic JOBCLASS statements that include STC/TSU that set batch only attributed

Fix

- JES2 z/OS 2.1

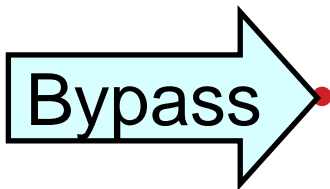


OA43333 – SAPI APPLICATION FAILS TO PROCESS NEW SYSOUT

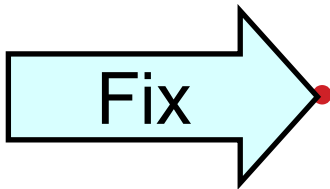


• SAPI application does not use ECB to discover new work

- Uses a timer to redrive requests
- New work not passed to application
- Restart application to select new work



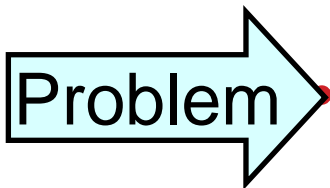
• Must restart application to select new work



• JES2 z/OS 2.1

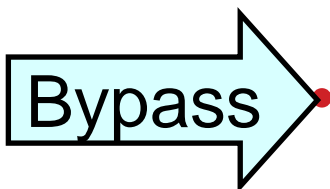


OA43123 – \$QXX ABENDS AFTER ISSUING A \$T CKPTDEF,MODE=DUPLEX

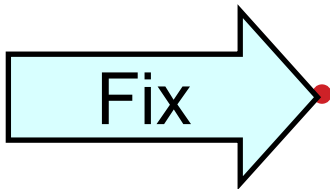


• Various error after a switched from DUAL to DUPLEX mode

- \$T CKPTDEF,MODE=DUPLEX
- CKPT changes can be lost and CKPT downleveled
- Impact may be delayed



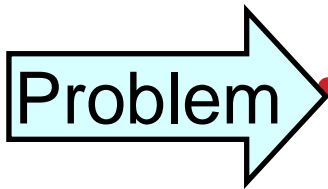
• Do not alter from DUAL to DUPLEX mode



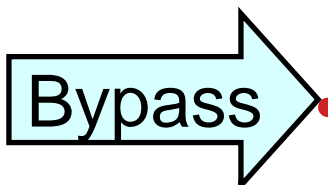
• z/OS 1.12, 1.13, 2.1



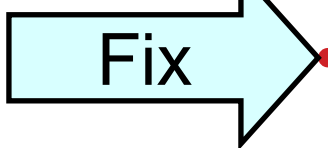
OA41571 – Various ABENDs after commands



- Command issued with null affinity operand
 - Eg \$TJ(*),/SYSAFF=()
 - SCHENV_AFF and SECLABEL_AFF affected
 - Also SRCVLASS QAFF=()
 - Can be \$T or \$D command
- Results can be varied and cross MAS
 - Checkpoint updates are lost
- Error can be greatly delayed after command



- Avoid null affinity operands



- JES2 z/OS 1.12 and 1.13
 - 1.11 with extended contract



OA42099/OA41738 – ENF performance

Problem

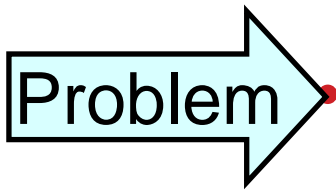
- Extra overhead listening to multi system ENFs
 - Overhead because JES2 listens to ENFs it issues
 - ENF filtering was done in listen exit
 - APARs add bit masks to ENF for filtering
 - Filtering done by ENF and not listen exit
 - If no listeners on member for ENF, then SRB is skipped
- OA41738 adds new bits to ENF being issued
 - **MUST be on all members (conditioning APAR)**
- OA42099 changes listener to filter on bits
 - If sender did not set bits, then EXIT does not run
 - SDSF (and others) displays break in this case

Fix

- JES2 z/OS 1.12 and 1.13
 - 1.11 with extended contract

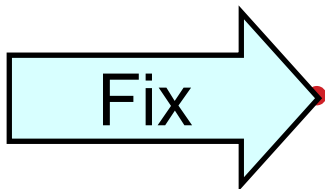


OA42996 – Reporting Class Incorrect



Reporting class not correct after service class change

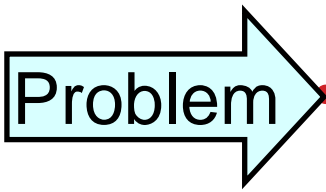
- PE of OA40963
- Service class changes AND other classification attributes
- Reporting class does not reflect the other changes
 - Never changes after service class change



JES2 z/OS 1.13

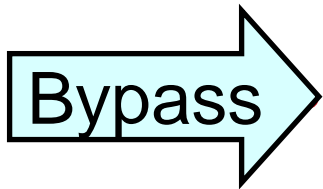


OA42787 – Duplicate job names running



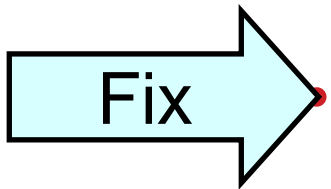
Jobs with duplicate job names running

- DUPL_JOB=DELAY is set
- Only for job classes S and T
- Only after command set DUPL_JOB=NODELAY for JOBCLASS STC or TSU
 - Could have be set as part of a generic set



Set DUPL_JOB=DELAY for JOBCLASS STC/TSU

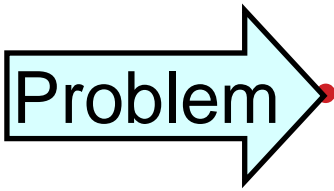
- Does not effect STCs or TSUs



JES2 z/OS 1.12 and 1.13

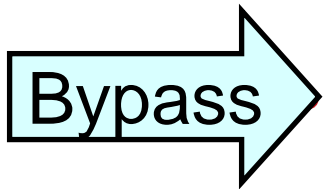


OA42452/OA43046 – Invalid Data from Device SSI

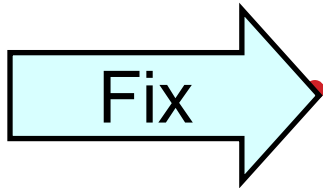


Init deck error encountered for a device

- Device is created but not completely
- Device SSI on another member can return bad data
- This bad data can cause SDSF (etc) ABENDs
 - Cannot process output from device SSI



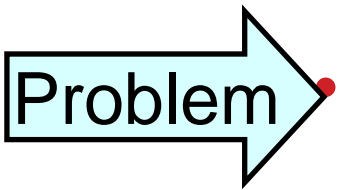
Fix errors in initialization decks



JES2 z/OS 1.12 and 1.13

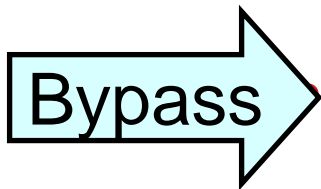


OA41793 – Loop processing free JOEs



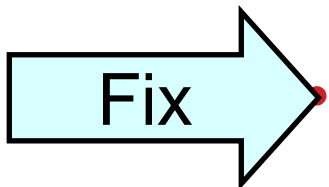
Command issued to free many JOEs

- Purge job command issued
 - \$PJQ, \$PJ, \$PT, \$PS
 - NOT \$POJQ, \$POJ, \$POT, \$POS
- Command need to run for more that 0.4 seconds
- Command does not hit DISPMAX and waits for CMBs
- Symptom can occur hours later



Ensure number of CMBs greater than DISPMAX

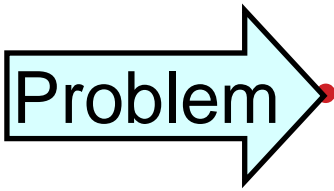
- Should be significantly larger



JES2 z/OS 1.12 and 1.13

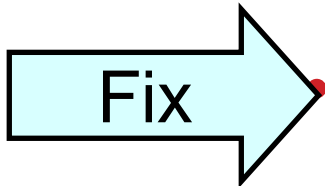


OA41768 – SAPI/PSO PCE Deadlock



• PSO held data set and SAPI deadlock

- Both accessing same held data set at the same time
- JOB lock and BERT lock involved in deadlock
- Requires hot start to clear problem
- Deadlock could be spread across members



• JES2 z/OS 1.12 and 1.13



OA41440 – FSS printer fails to restart

Problem

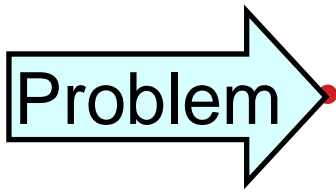
- FSS device fails or is canceled
- Address space could be gone
- Restart of FSS device fails
 - IPL needed to clear this
- Error recovery left FSS in a non-restartable state
- **IF YOU RUN FSS, YOU WANT THIS!**

Fix

- JES2 z/OS 1.12 and 1.13

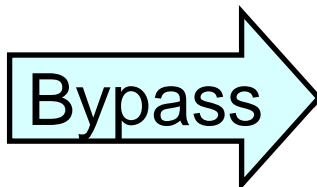


OA41330 – Wild Branch after ABEND

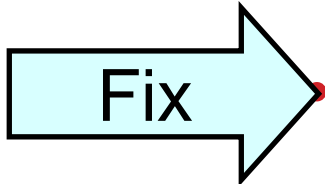


Address space is canceled or ABENDs

- Occurs while a JCT write is in progress
- SPOOLDEF CYL_MANAGED=FAIL set
- Code to clean up JCT takes a wild branch
- Cleanup for the address space does not complete



Switch to CYL_MANAGED=ALLOW



JES2 z/OS 1.12 and 1.13



OA40901 – LINEx.ST improper JQE update



Problem

- NJE SYSOUT transmitter encounter error
 - Decides to hold job
- Job is active on another process (OUTPUT)
- NJE update does not have proper serialization
 - \$Q14 ABEND results
- Rare timing window



Fix

JES2 z/OS 1.12 and 1.13



OA40790 – \$SJB chaining error cause IPL

Problem

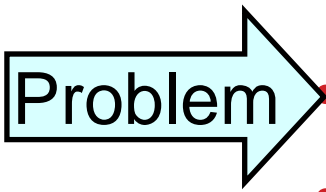
- Unknown overlay of JES2 HASB data area (CSA)
- Recovery of HASB results in \$SJB removal
- But running address space still using \$SJB
 - Double usage of \$SJB is possible
- Various chaining problems can result
 - Loop in \$SJB chain can require IPL to fix
- APAR correct recovery code and prevents loop
 - Does not address overlay

Fix

JES2 z/OS 1.12 and 1.13



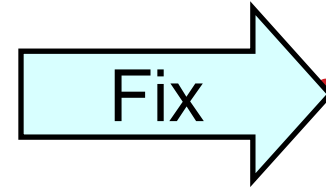
OA40755 – CKPT reconfiguration hangs



- CKPT on CF is being reconfigured
- During reconfiguration process, CF rebuild is done
 - Any system managed process could trigger this
- Reconfiguration ends up waiting forever for subtask



• System managed process during CKPT reconfiguration



• JES2 z/OS 1.12 and 1.13



OA40735 – ABEND in CKPT reconfiguration

Problem

- Checkpoint on DASD
 - I/O error reconfiguration occurs due to lost paths
 - Device old CKPT is on cannot be accessed
- CLOSE processing for old data set fails
 - ABEND314 RSN4
- JES2 terminates

Fix

JES2 z/OS 1.12 and 1.13



Don't Run Out of BERTs

- Don't run out of BERTs



Questions?

Questions?

Session 14661



Mainframe 50 April 7th 1964 - April 7th 2014
<http://www.ibm.com/mainframe50/>





OA39972 – ABEND submitting job to RJE RDR

- Problem** → Invalid job submitted via RJE (or card) RDR
- Message \$HASP125 SKIPPING FOR JOB CARD
 - Other invalid jobs will get an ABEND 0C4
 - 0C4 continue until valid job is submitted

Fix → JES2 z/OS 1.13



OA39891 – Segmented or spun output errors

Problem

• Job creates SYSOUT with SEGMENT= value
or

• SPIN data is spun (due to SPIN any support)
and

• Data set is referenced by OUTPUT cards
then

• If first segment prints and purges, then subsequent
segment cannot be printed (\$CBIO error)

Fix

• JES2 z/OS 1.13



OA38935 – ENF 58 and 70 Fixes

Problem

Excessive multi system ENFs some with no useful data

- PSO issued ENF 58 when not requested
- JQS issued ENF 70 in cases where not needed
- Missing data in various ENF 58 and 70
- See APAR for list of issues fixed
- Added ENF 58 (43 & 44) and 70 (45 & 46) JES2 \$TRACE ids
 - Send and receive ENFs respectively

Fix

JES2 z/OS 1.11, 1.12, 1.13



OA37835 – Track Group Recovery Improvements



Problem

- Unusual problem in disaster recovery
 - Seems to be related to PPRC and Flash Copy of SPOOL
 - Signature records (SPOOL ownership records) corrupted
 - Causes double allocation of SPOOL at target (recovery) site
- APAR makes SPOOL garbage collector more conservative
 - Examines data and signature record before reclaiming space
 - Could delay SPOOL recovery
- Original problem has been diagnosed and fix is in process
 - Fix number 264749, microcode release 6 and 7



Avoid!

Flash Copy and PPRC of SPOOL



Fix

JES2 z/OS 1.11, 1.12, 1.13



The Perfect Storm

- JES2 **RED ALERT** – Draining and halting SPOOL volumes
- OA36256 – Fixed a problem with JESXCF messaging
 - Messages went to wrong member name due to down level QSE
 - Changed tests so a \$QSUSE was done to wait for latest QSEs
 - Delayed XCF member status information
 - Initializing member appears MVS-GONE for one JES2 dispatcher cycle
 - Exposes bug in HASPSPOL that messes up DASALOCS flags
 - Only for starting, draining and halting volumes
 - Members allocated to a CKPT volume
 - Causes HASPSPOL to make incorrect decisions
 - Thinks volume can be halted/drained prematurely
 - Real problems if/when a halting/draining volume is started (\$S)
 - DYNALLLOC fails if a members DASALOCS bit is wrong



The Perfect Storm

- OA38016 – Fixes obscure problem when \$S command fails
 - Only happens if \$S command and some members need to allocate
 - “Should” never happen for draining volume
 - Rare for halting volume
 - If allocate fails, volume is REMOVED from the configuration
 - All data on the volume is lost
 - Problem was fixed in z/OS 1.13, APAR was rolling down fix
- Second problem with DEB for SPOOL volumes
 - If volume incorrectly drained or halted (due to OA36256 and \$S problem) the DEB is not cleared
 - New volume starting on member will use OLD DEB and OLD VOLUME
 - Only on members affected by OA36256
 - Member using different physical volumes for same SPOOL
 - Also addressed by OA38016



The Perfect Storm

- OA39737 – Corrects XCF status problems
 - Removes \$QSUSE for status updated
 - Member status is now blend of XCF status and \$QSE status
 - Corrects problems causing DASALOCS to be bad
 - Adds DEBUG option to track member status changes via WTO
- What to do?
 - Back off OA36256 if possible
 - Apply OA38016 as soon as possible (fixes data loss problem)
 - Avoid Halting (\$Z) or draining (\$P) SPOOL volumes
 - If you want to drain, set SYSAFF=(-ANY) and wait for jobs to purge
 - Do \$P when all jobs are off volume
 - If a \$S fails with
IKJ56246I DATA SET SYS1.HASPACE NOT ALLOCATED, FILE IN USE
then hot start the member to correct the DASALOCS bits
 - Warning, JES2 can halt volumes at any time due to loss of connectivity



OA39337 – JES2 start fails with \$HASP401

- In z/OS 1.13, the IHADVA size unintentionally increased from x'18' to x'20' bytes
- OA37764 restored the original x'18' size of the IHADVA
- JES2 is sensitive to the size of the IHADVA
 - Used in data areas shared across modules
- JES2 PTF all used larger size
- Local re-assemblies could use smaller size if OA37764 applied
 - Caused inconsistent mappings resulting in errors and failed starts
- Overabundance of caution resulted in all PTFs shipping parts using IHADVA to be PEd.
 - Fix reworks code to not have dependency across modules
- Not a problem if you do not assemble JES2 locally or if you always assemble everything when anything changes



Performance APARs

- OA38935 – Reduce number of ENF70s and ENF58s
 - Also corrected problems with payload in ENFs
 - Added \$TRACE points to trace ENFs sent and received
 - Useful if developing applications to use ENFs
- OA36382 – Eliminate extraneous job lock to improve SAPI performance
 - Major improvements if you have many SPIN data sets and multiple SAPI threads
 - PE – Output not selected after SAPI splits JOE – OA40499
- OA36328 – \$TO commands to change DEST= can take a long time due to unnecessary subtask processing



DOC APAR OA38750

- \$HASP492 message indicates start process status
\$HASP492 JES2 MEMBER-N1M1 QUICK START HAS COMPLETED
- APAR OA24118 dealt with ABENDs during \$E MEMBER
 - ABENDs caused the \$E MEMBER processing to fail
 - New variant of \$HASP492 was issued in this case
\$HASP492 JES2 MEMBER-N1M1 RESTART HAS FAILED
 - Variant is ONLY for RESTART (\$E MEMBER) processing
 - Other starts just fail to complete and JES2 terminates
- Message update appeared in recent manuals
 - Not clear when HAS FAILED came out
 - Was a concern for automation
- DOC APAR clarified the message and variants
- Ironically, automation was part of the reason for using the same message ID
 - It indicated that the process completed



OA39223 – JQE Overlay Prevents \$ZAPJOB

Problem

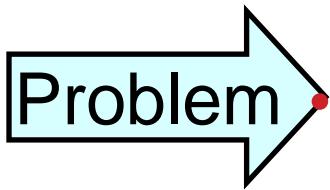
- Unknown problem (exit?) overlays JQE node number
 - Input (JQEINPND) or Execution (JQEXEQND) node
- Various ABENDs trying to access job
- Attempt to \$ZAPJOB fails with similar ABEND
- \$ZAPJOB should not fail
- Code added to validate binary NJE node before using

Fix

JES2 z/OS 1.11, 1.12, 1.13



OA38951 – JES2 CKPT hung after System managed process



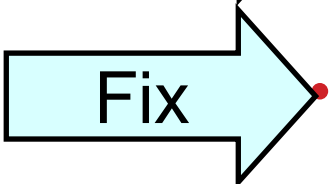
Problem

- The JES2 CKPT process (PCE) can hang
 - CKPT must be on CF
 - A system managed process has just completed
 - Such as a rebuild
 - The post of the CKPT on CF subtask is lost
 - CKPT PCE waiting for subtask, subtask waiting for PCE
 - Very small timing window
- No CKPT work will happen (Start, logon, etc)
 - Hot start clears problem



Avoid!

- System managed rebuild of CKPT CF structure



Fix

- JES2 z/OS 1.11, 1.12, 1.13 - OPEN



OA38043 – Allow Null JOBCCLASS QAFF=

Problem

• Can remove members from QAFF but list cannot be null

- Wanted to shut down last member in QAFF but had to hold the jobclass instead of removing member
- No good reason to disallow null list
- Code that ensures list not null removed

Fix

• JES2 z/OS 1.11, 1.12, 1.13



OA38683 – \$JOA Cleanup Problem

Problem

- PCE ABENDs owning BERT lock for a \$JOA
- Code in MISC PCE attempts clean up BERT lock
 - Uses wrong offset (offset from JQE) for BERT token
- Result is various errors including \$BR3
 - Errors could result in corruption of some existing BERTs
- Problem in z11 mode only
- If you encountered the problem, expect to see errors on warm start (even with APAR on)
 - \$DISterr at Label BERTERR
 - \$HASP483 JES2 JOB QUEUE ERROR, RC=51

Fix

JES2 z/OS 1.11, 1.12, 1.13



OA38671 – CPU Loop after \$CJ

- Problem**
- Loop in JES2 converter processing
 - Caused when job is \$CJed while in converter
 - \$CJ has to occur at an inopportune moment
 - Middle of JES2 chaining an SDB for a data set
 - Loops occurs due to additional validation during free

- Avoid!**
- \$CJ of jobs in conversion (unless truly hung)

- Fix**
- JES2 z/OS 1.13



OA38135 – SYSLOG Browse Hang (SDSF)

- Problem** → Loop can occur if browsing SYSLOG as it is SPUN
- Easy to re-create if writelog done from SDSF browse panel
 - TSO user in a never ending loop with an I/O in it

Avoid! → WRITELOG from SDSF SPOOL Browse panel

Fix → JES2 z/OS 1.11, 1.12, 1.13

