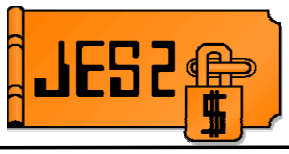
JES2 Product Update: z/OS 1.7

SHARE 105, Session 2655
Tuesday, August 23, 2005

Chip Wood
JES2 Design/Development/Service
Poughkeepsie, NY
chipwood@us.ibm.com

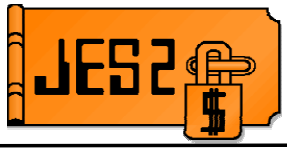
JES2 z/OS 1.7



- **New changes in z/OS 1.7**
  - NJE over TCP/IP
  - Large (>64K track) SPOOL Data Sets
  - Reader/NJE exit changes
  - Long SYSIN support (32K LRECL)
  - Table pair enhancements
  - $SCAN from non-main task environments
  - SSI for JES2 monitor information
  - SAPI and extended status enhancements
  - Checkpoint recovery (DAS corruption)
  - Other goodies (requirements!!!!)
  - Drop of support for release 4 mode

JES2 z/OS 1.7 is the largest and most function-rich release of the product in many years. This session will provide an overview of the function added in the release. Many of the functions will be discussed in much more detail in sessions throughout the week.

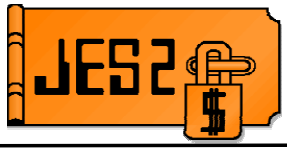
JES2 z/OS 1.7 Installation



- **From JES2 OZ/390 R10 or earlier**
  - Migrate to more recent spool-compatible release first (z/OS 1.5) to avoid cold start
  - **$ACTIVATE,LEVEL=z2** on that release (no $ACTIVATE support in z/OS 1.7)
- **From JES2 z/OS 1.2**
  - **$ACTIVATE,LEVEL=z2** required to avoid cold start
  - No MAS coexistence (all member warm start)
- **From JES2 z/OS 1.4 or 1.5**
  - **$ACTIVATE,LEVEL=z2** required to coexist with z/OS 1.7
  - APAR **OA08145** needed on z4/z5 member
    - Includes toleration for:
      - Long sysin records
      - Local node name changes
      - Persistent NJE connections
      - $HASP549 message changes
    - **OA11953**, **OA12472** fix errors in OA08145

To migrate to JES2 z/OS 1.7 via warm start, you must be $ACTIVATEd at the z2 level.

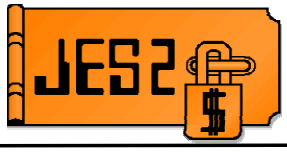
JES2 z/OS 1.7 Installation



- **New JES2 distribution library**
  - Moved from **SYS1.SHASLINK** (PDS) to **SYS1.SHASLNKE** (PDSE)
    - Allows HASJES20 to be a split load module
    - Most JES2 modules are now loaded above 16M
      - 1.2M of code was moved
      - A few modules had to stay below 16M
        - Some code ($SAVE, $RETURN, etc) has to tolerate AMODE 24 entry
        - Appendages, etc. that must reside below 16M (HASPBSC, HASPPRPU)
        - Extensive use of AL3 for addresses (HASPSNA, HASPCOMM)
  - **Check your JES2 proc!!!!!!**
- **If you do your own assembly/linkedit of JES2**
  - JES2 modules should be assembled with the **XOBJECT** and **LIST(133)** parameters
  - JES2 should be linkedited into a PDSE with the **RMODE(SPLIT)** option

A significant change to the packaging of JES2 – the load library is now a PDSE. This was done to take advantage of the RMODE(SPLIT) binder option to load most of the HASJES20 load module above the 16M line. The new library is **SYS1.SHASLNKE.**

Checkpoint Recovery

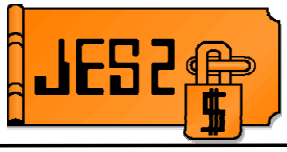


- **Problem:**
  - Checkpoint data may be corrupted, resulting in an outage.
  - Checkpointed control block, representing a spool volume ($DAS), may be corrupted as a result of overlays.
  - When a $DAS is corrupted, its spool volume can become unusable and result in data loss.
- **Solution:**
  - Each $DAS and appropriate chains are verified and rebuilt, if necessary, on any warm or hot start.
  - If a rebuild was necessary, the operator will be notified via the $HASP896 message.

Because of the nature of the JES2 checkpoint, any corruption of the JES2 checkpoint could potentially cause a sever outage (possibly even a cold start). Over the years, we have added code to deal with corruption of the JQEs and JOEs and to recover as much as we can. In SP 5.2.0, the “sniffer” was added to deal with corruption of the track group maps. The only remaining checkpointed area whose corruptuion could lead to extensive data loss is the DAS, which represents the spool volumes themselves.

Additional validation has been added in z/OS 1.7 to detect and correct problems with the DAS CTENT, minimizing the amount of damage that can occur if this area is corrupted.

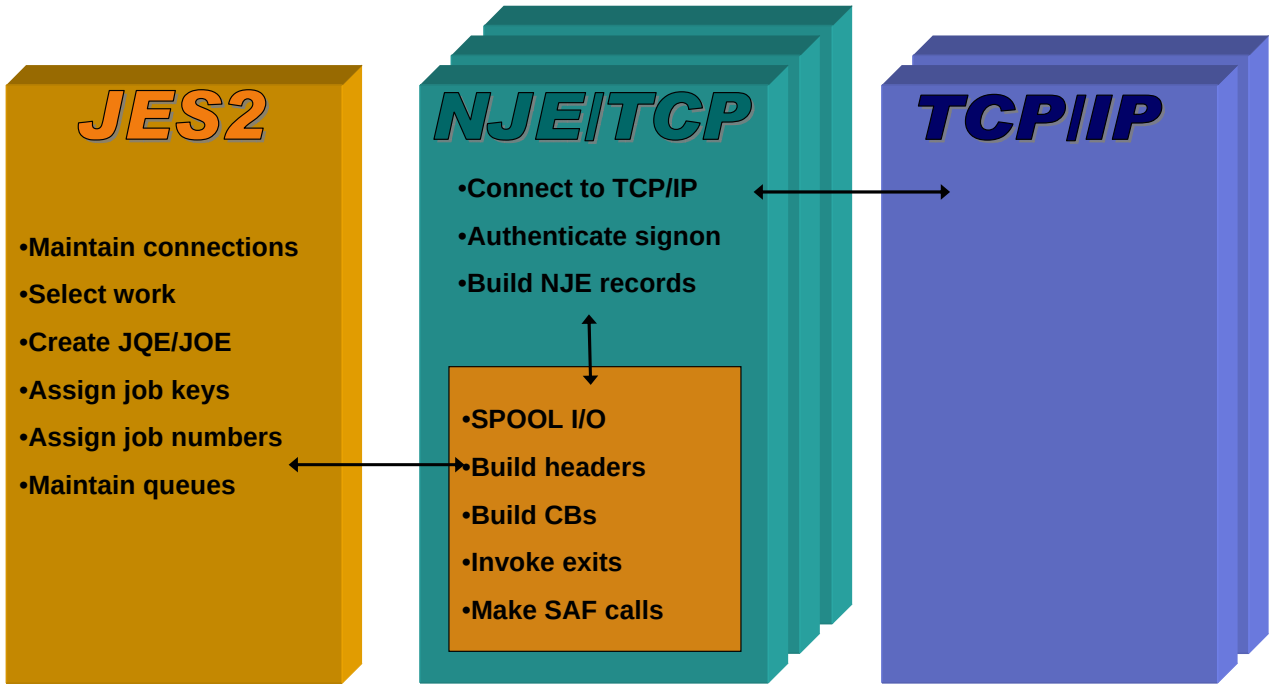
NJE over TCP/IP



- **Currently JES2 supports NJE over SNA and BSC networks**
- **TCP/IP is the standard for networking today**
- **VM (RSCS), AS/400, and VSE/POWER all have supported NJE over TCP/IP for years**
- **JES2 z/OS 1.7 supports established TCP/NJE protocol**
  - Modified BSC CTC protocol
  - Enabled by APAR **OA12364**, available **1Q2006**
- **Enhancements to protocol also included for**
  - IPv6 connections
  - Enhanced security
  - Large LRECL SYSIN data streams
- **Overall better RAS characteristic have been designed in**
  - Fewer outages than current NJE
  - Better performance
- **Session 2658, JES2 for z/OS 1.7 Large Spool, NJE over TCP/IP, and Other Features, Wednesday 3:00**

Support for NJE over native TCP/IP has been one of the most requested JES2 enhancements over the last several years.

NJE over TCP/IP *(Cont…)*



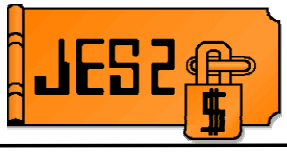
One of the primary concerns customers have had with JES2 address space outages is that when the JES2 address space ABENDs, all NJE connections are lost. Even though the JES2 address space can be restarted, the NJE connections must also be reinitialized. Many customers have automation to do this, but only when the system is IPLed. So, the JES2 address space outage turns into a system outage.

In order to address this, the NJE/TCP support is moved to a separate address space, where the connection can remain active when the JES2 address space is unavailable. This approach has multiple benefits:

1. Availability – Outages of the JES2 address space do not affect the availability of the NJE connection. Conversely, problems in the NJE address space do not result in a JES2 outage as they would have in the past.
2. Performance – Most of the work associated with the NJE connection (I/O, building headers and trailers, etc.) is being done outside the JES2 address space, rather than under the JES2 main task. This frees up cycles in the already overtaxed JES2 main task to do other things.

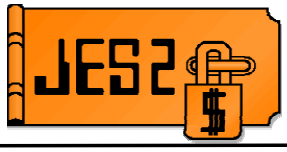
The actual communication with TCP/IP is being done via a new common JES2/JES3 component, IAZNJTCP.

NJE over TCP/IP *(Cont…)*



- **Protocol is that originally developed for VM**
  - Assigned port is 175, port name is VMNET
  - Originally described as BITNET II protocol
  - Documentation is in NJE formats and protocols
    http://publibfp.boulder.ibm.com/cgi-bin/bookmgr/BOOKS/iea1m503/6.5
  - Essentially BSC CTCA protocol with TCP/IP wrappers
  - Support for IPv6
  - Support for SSL/TLS
  - Support for multiple stacks, VIPA, Sysplex distributor
  - Binary IP addresses now optional in data records
    - Better support for NAT
  - Stronger authentication available (using SAF/RACF)
    - Controlled by **NODE*nnn* SIGNON=SECURE|COMPAT**

NJE over TCP/IP *(Cont…)*



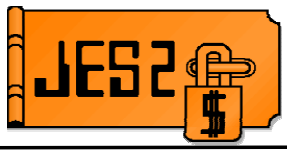
- **Within JES2, the externals will parallel what was defined for SNA**

SNA	NJE/TCP	Description
LOGON	NETSERV	Represents a NJE/TCP address space
APPL	SOCKET	Maps an NJE node name to a TCP/IP address (either explicit or a name)
LINE	LINE	Logical connection

- Commands to start and stop networking/devices similar to SNA
  - **$SLINE**, **$SNETSRV**, **$SN**, **$ELINE**, **$PLINE, $ENETSRV**, **$PNETSRV**, etc.
- NETSERVs can bind to all defined IP addresses or a specific address

The externals for JES2 NJE over TCP/IP are modeled after the SNA externals, with a NETSRV corresponding to a LOGON and a SOCKET corresponding to an APPL. Logical lines specifying UNIT=TCP are used.

NJE over TCP/IP Externals



- **NETSRV statement (NETSERV or NSV)**

```
NETSRV1     STATUS=DRAINED,SOCKET=,STACK=,
            TRACEIO=(JES=NO,COMMON=NO,VERBOSE=NO)
```

- **SOCKET statement**

```
SOCKET(LOCAL)      IPADDR=*LOCAL,PORTNAME=VMNET,
                   SECURE=NO,LINE=0,NODE=2,REST=0,
                   NETSRV=0
SOCKET(OTHER)      IPADDR=9.117.234.95,PORTNAME=VMNET,
                   SECURE=NO,LINE=0,NODE=82,REST=0,
                   NETSRV=0
```

- **LINE statement**

```
LINE12      UNIT=TCP,STATUS=DRAINED,LOG=NO,
            PASSWORD=(NOTSET),REST=0,TRACEIO=(JES=NO,
            COMMON=NO,VERBOSE=NO),JRNUM=DEFAULT,
            JTNUM=DEFAULT,SRNUM=DEFAULT,STNUM=DEFAULT
```

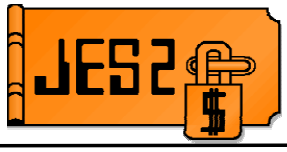
The NETSRV statement (which may also be specified as NETSERV or NSV) defines the characteristics of the network server.

The SOCKET statement defines the IP address, port, and other attributes associated with one end of a TCP/IP connection. These definitions are used in 2 places:

1. The SOCKET= parameter on the NETSRV statement identifies the IP address and port associated with the local node (i.e. this end of the connection)
2. The $SN,S= command identifies the socket representing the IP address and port we wish to connect to (i.e. the other end of the connection).

The LINE statement defines a logical line. For TCP/IP connections, specify UNIT=TCP.

NJE NODE Definition Changes

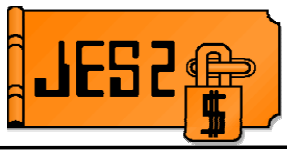


- **NJE node definitions are now MAS scope**
  - Changes on one member reflected on others
  - Data is stored in the JES2 checkpoint
  - Applies to **NAME=**, **SUBNET=**, **PATHMGR=**, **ENDNODE=**, **PRIVATE=**, and **DIRECT=**
  - Init deck changes accepted when other members active
    - Warning message if incompatible, option to continue or terminate
- **Change node names when node is active (no restart)**
  - **$TNODE(*nodename*),NAME=*newname***
    - Indicates node is renamed, not redefined
  - Applies to local node name as well (was all-member warm start)
- **Change maximum node number via command**
  - **$T NJEDEF,NODENUM=** (was all member warm start)
  - Value can only be increased

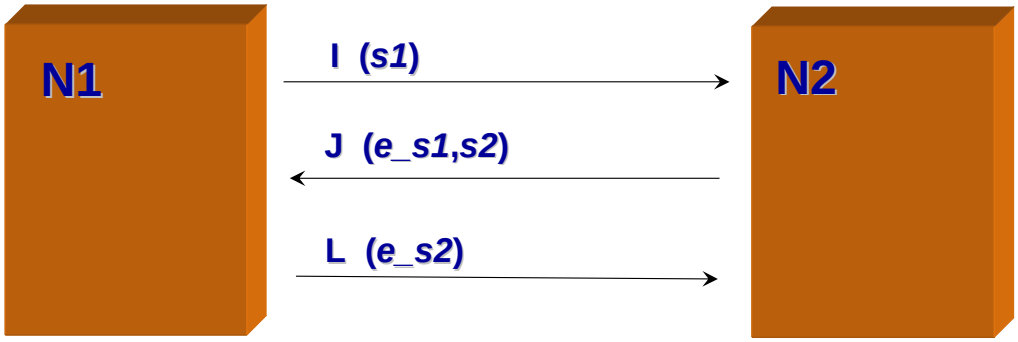
A potential cause of problems within a MAS lies with the NJE definitions. It is possible, prior to z7, to bring up a MAS where the node definitions are inconsistent from member to member, or inconsistently changed across a JES2 restart. Problems have even surfaced acrossed planned changes, where MAS members temporarily have inconsistent definitions across a rolling MAS-wide warm start.

To provide consistency in these cases, several key attributes of node definitions are now stored in the checkpoint, where they can be shared across the MAS and checked for consistency on a warm start. A fallout of this change is the ability to support the dynamic change of the maximum number of nodes and the local node name (if necessary). These two changes previously required an all-member warm start.

NJE Security Changes



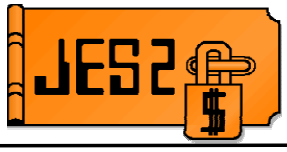
- **Support for SSL/TLS to be included in NJE over TCP/IP**
  - Controlled by **SECURE=YES/NO** parameter on **SOCKET** statement
  - Uses Application Transparent TLS (AT-TLS) API
- **Secure form of NJE signon now supported**
  - Exchanges DES-encrypted passwords in I/J signon records
  - Controlled by **SIGNON=SECURE|COMPAT** on **NODE** statement
  - Uses **APPCLU** class in RACF/SAF
    - Entity is **NJE.*node1.node2***
    - Uses **SESSKEY** associated with profile for encryption
  - Can be used by SNA or BSC nodes as well as TCP/IP



In the TCP/IP world, security becomes much more important than in the more protected environments of SNA and BSC. TLS and SSL go a long way, but additional validation may be required. TLS and SSL, for example, help ensure that the partner on the other end of the connection is who he claims to be in a TCP/IP sense. However, we still need to validate that he is who he says he is in an NJE connectivity sense. Also, if TLS/SSL is not available, you may not want to send nodal passwords into TCP/IP-land in clear text.

To improve this security, we now allow for DES-encrypted passwords to be sent in NJE connection records in place of the current clear text values.

Reader/NJE exit changes



- **TCP/IP NJE processing occurs outside the JES2 main task**
- **INTRDR processing also moving outside JES2 main task**
- **Main task exits no longer get control for TCP/NJE and INTRDR jobs**

- **New exits have been added corresponding to current exits**
- **Additional main task exits have been defined**
  - When jobs are added to the job queue

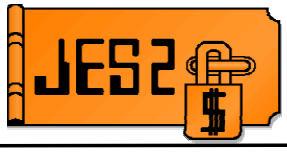
- **Exits 36 and 37 are still called, but in some cases, from a different address space**
- **Exit 8 is called in some cases instead of exit 7**

The changes to NJE to implement TCP/IP will make it impossible to call the traditional HASPRDR exits in the JES2 main task. Similarly, changes to internal reader processing will also make it impossible to call the traditional HASPRDR exits in that environment. To address this, a new set of input processing exits has been defined. These exits will run in the user environment in the NETSERV address space. In addition, new exits will be defined in the main task when jobs are added to the job queue.

In the case of exits 36 and 37, the exits will still be called, but they are called from a different address space.

For control block I/O, since the I/O is being done outside the main task, exit 8 instead of exit 7 will be called.

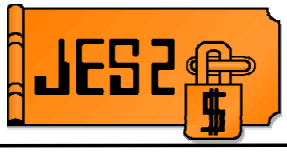
Reader/NJE exit changes *(Cont…)*



- **New exits have been added corresponding to these main task exits**
  - **2** – JCL job card
  - **3** – Job card accounting field
  - **4** – JCL and JES2 control (JECL) statement
  - **20** – End of input
  - **39** – NJE SYSOUT SAF rejection
  - **46** – NJE header transmit exit
  - **47** – NJE header receive exit
- **New exits have been defined for**
  - Move JQE to next phase ($QMOD)
- **Other exits affected:**
  - **7**/**8**– Control block I/O
  - **13** – TSO/E NETMAIL notify (deleted)
  - **36**/**37** – Pre and post SAF exit

This is a list of the exits affected. New exit numbers were defined for exits that need to be called outside the main task.

Reader/NJE exit changes *(Cont…)*

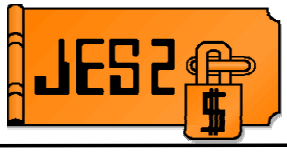


New Exit	Similar exit	Environ	Function
50	**20**	USER	End of input
51	*****	JES2	$QMOD (job phase change)
52	**2**	USER	Input processing - JOB card
53	**3**	USER	Input processing - Accounting field
54	**4**	USER	Input processing - JCL/JECL
55	**39**	USER	NJE SAF rejection
56	**46**	USER	NJE header/trailer transmit
57	**47**	USER	NJE header/trailer receive

- All exits (new and changed) will be passed XPLs
- XPLs for new and similar exit will be the same
- New data areas will contain former PCE/DCT fields
  - Passed to both exits
- Old exits will be passed same data as in previous releases

This is a list of the new exit numbers, the similar old exit, and the environment of the new exit. All exits will be passed XPLs. Existing exits will have XPLs available as well as the current input registers. The XPLs for the new and old exits will have the same data (but separate mappings). Some data areas that were in PCEs will be moved to new data areas that will be common to both environments. The XPL will formalize some of the interfaces and simplify some of the tasks commonly performed in each exit (based on customer and vendor feedback).

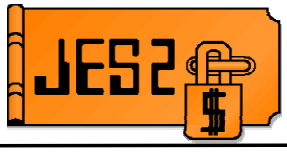
Reader/NJE exit changes *(Cont…)*



- **Old style exits (2, 3, 4, 20, 39, 46, 47) still used for:**
  - Local card readers
  - RJE readers
  - SNA and BSC NJE transmitters and receivers
  - Spool Offload transmitters and receivers
- **New style exits (52, 53, 54, 50, 55, 56, 57) used for:**
  - Batch Internal readers
  - STC and TSU internal readers
  - TCP/IP NJE transmitters and receivers
- **New exit 51 receives control for all phase changes:**
  - Job moves from $INPUT to $XEQ, etc.
  - Job requeued for execution

The old exits are still used for all but internal readers and NJE/TCP. The new exits are used for NJE/TCP and internal readers. Exit 51 is a main task exit that gets control as jobs move from one phase to the next. For NJE/TCP receivers and internal readers, this is the first main task exit for the job.

Reader/NJE exit changes *(Cont…)*

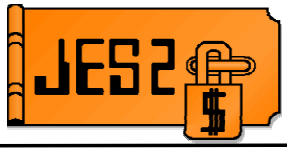


- **$EXIT 13 has been deleted**
  - $HASP548/$HASP549 no longer issued from SYSOUT receiver
    - Too early anyway
    - No way to force for locally created output
  - $HASP548/$HASP549 now issued during OUTPUT processing
    - When job is actually available to TSO RECEIVE
    - Can now force message out for locally created output
    - Can use exit 13 in down-level releases to simulate this behavior
      - HASX13A with OA08145
- **$EXIT 13 functionality replaced by:**
  - **NJEDEF MAILMSG=YES/NO** (since SP 4.3.0!!!)
  - $EXIT 47 (SYSOUT receiver DSH reception)
    - Process data set header fields in SYSOUT receiver
  - $EXIT 40 (JOE creation)
    - Flags in $XPL to influence whether messages issued
    - Flag in PDDB to influence job eligibility for message
    - Can use exit 13 in down-level releases to move notify message to $EXIT 40 time
      - HASX13A with OA08145

In this release, JES2 is addressing a long standing complaint about the message issued when SYSOUT for a TSO user is received. Prior to this release, the message was issued early in processing the SYSOUT data set that was being received. If the SYSOUT data set was large and the TSO user did a receive after seeing the message, it is possible that the data set may not yet be available for processing.

The notify processing was moved from SYSOUT reception processing to OUTPUT/SPIN processing. This ensures that the message is not received before the output is ready for processing. However, exit 13 no longer made sense in this environment. As a result, and because it is unlucky, we deleted exit 13. The function of the exit can be replaced by the existing external NJEDEF MAILMSG and new function added in exit 40.

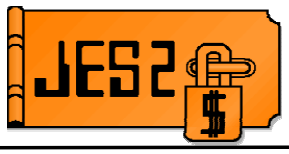
Reader/NJE exit changes *(Cont…)* *What can I do right now????*



- **Familiarize yourself with current exits**
  - Exits 2, 3, 4, 20, 39, 46, 47
    - VERY commonly used exits
      - If you use internal readers, you WILL need to change them soon.
    - Check for main task dependencies
    - In general, updateable JQE and JCT will be available for reader/receiver exits in user environment
  - Exit 13
    - Use **NJEDEF MAILMSG=** or Exit 40 to influence message issuance
    - Use Exit 47 to manipulate fields based on dataset header
  - Exit 7, 8, 36, 37
    - Understand reason for exit
      - For example, last write of JCT under reader PCE
  - Determine whether exits are still required

- Session 2665, JES2 Exits Overview, Thursday, 1:30
- Session 2656, JES2 Migration, Wednesday 1:30

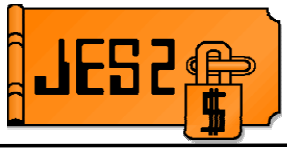
JES2 and Rel 7 Migration Offering (fee)



Have an IT specialist update your exits

US contact is **John Hock/Phoenix/IBM**
IBM Global Services
Certified Consulting IT Specialist
zSeries Delivery Team
Phone: 602-217-2206,
Elsewhere, ask Chip Wood

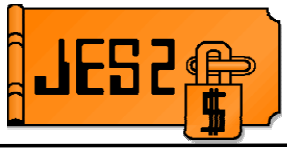
Table pair enhancements



- **Problem: Table pairs often require fields in $UCT data area**
  - Only one $UCT per installation
  - Only one $EXIT 0 per installation
  - Difficult for multiple vendors to share $UCT
  - Up to installation to merge multiple vendor exit 0 routines or $UCTs
- **Solution: Create alternatives for UCT fields**
  - ADCON or VCON
  - Named token pointer to data area
    - Can be TASK, HOME, PRIMARY, SYSTEM level
    - Also SUBSYS level (SYSTEM level, last 4 chars of name set to subsystem name)
    - CB address assumed in first 4 bytes of token
  - New $TOKENSR service to create, retrieve, or delete
- **In addition, use dynamic exit service to allow multiple exit 0s**
  - Exit name is HASP.$EXIT0
  - Old exit 0 implementation remains
  - Parameter list is unchanged

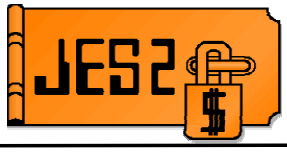
A long standing problem with vendors has been the single EXIT 0 in JES2. Since EXIT 0 is invoked very early in JES2 initialization, installations had no control over what exits run at this time (except to code or link edit router exits). To address this limitation, JES2 is using the dynamic exit services to invoke multiple exit 0 routines. The parameters needed to set up these exits is listed. One concern is that of tables ($SCANTABs for instance) that may exist in one of these exit 0s. The problem is JES2 has no control over when these exits are refreshed. If an exit 0 is refreshed, you could end up with tables in JES2 pointed to the freed storage for the old exit. To prevent this, IBM recommends that any dynamic exits be $MODLOADed from the dynamic exit 0 into non-dynamic storage. JES2 ships a sample exit 0 that uses this technique.

Table pair enhancements *(cont…)*



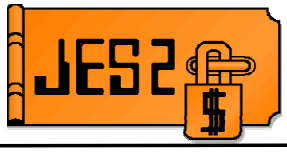
- **Typical new operands values**
  - OPERAND=(*field*,HCT) – value in HCT (existing option)
  - OPERAND=(*field*,UCT) – value in UCT (existing option)
  - OPERAND=(*label*,ADDR) – Obtain value via ADCON or VCON
  - OPERAND=(*offset*,TOKEN) – value in data area pointed to by token
    - OPERANDTK=(*name*,SYSTEM) – token at system level
    - OPERANDTK=(*name*,SUBSYS) – token at subsystem level
    - OPERANDTK=(*name*,HOME) – token at address space level
    - OPERANDTK=(*name*,TASK) – token at task level
- **Token not valid on all operand (examine each for options)**
- **First 4 bytes of token assumed to be control block address**
- **$SCANTAB CB= has similar options but different syntax**

Table pair enhancements *(cont…)*



- **Updated macros and operands**
  - $DCTTAB
    - CHAIN=, CHAINTK=, COUNT=, COUNTTK= PCEPTR=, PCEPTRTK=,
  - $DTETAB
    - EPLOC=, HEAD=, HEADTK=
  - $PCETAB
    - CHAIN=, CHAINTK=, COUNTS=, COUNTTK=, ENTRYPT=
  - $PCTAB
    - ENTRYPT=, RECOVPT=
  - $SCANTAB
    - CB=
  - $WSTAB
    - DEVCB=, MODSCB=

$SCAN from non-main task environments

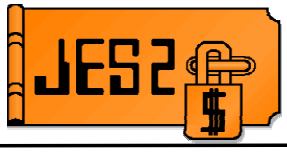


- **JES2 $SCAN services are now available outside the main task**
  - USER, SUBTASK and FSS environment supported
- **Includes $BLDMSG service**
- **Can be used from new input service exits to parse JECL cards (for example)**
- **User and dynamic tables supported**
  - Tables must be in CSA if called from user environment
  - CCTMGTP table pair in HCCT for dynamic BLDMSG tables outside JES2 address space
- **Many new $SCANTAB functions allowed**
  - CB=HCCT, CB=DTE, CB=(TOKEN,name)
  - CONV=NUMS (signed), CONV=NUMU (unsigned)
    - Many parameters now allow a max value of 4G instead of 2G
  - CONV=NUMT (format with thousands separator)
  - Four digit MSGID= values

The JES2 $SCAN services have been updated to support being called from outside the JES2 main task which includes being called from user environment exits and the use of the $BLDMSG services. This allows the $SCAN services to be used to parse basic JECL statements from the new input services exits in the USER environment.

The updated services support dynamic tables pair processing. Separate tables exist for user environment user of the $SCAN service (as well as the $BLDMSG service).

SSI for JES2 monitor information

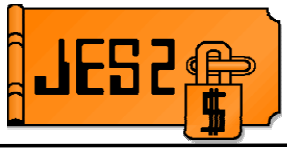


- **New function on the JOB information SSI (SSI 71)**
- **2 new functions added to IAZSSJI**
  - Get monitor information and return storage
- **New data area (IAZMOND) for new function**
- **Returns all information available via monitor commands**
  - Resource usage statistics
  - Main task CPU statistics
  - JES2 ERROR statistics
  - Main task WAIT statistics
  - JES2 Alerts
  - JES2 Notices
  - JES2 Tracks
  - Monitor status information
- **Some additional data returned**

- **Session 2658, JES2 & JES3 z/OS 1.7 SSI , Wednesday 3:00**

JES2 now provides the ability to obtain JES2 monitor information via a new call to SSI 71. This will be discussed in more detail at Session 2658.

SAPI and Extended Status Enhancements

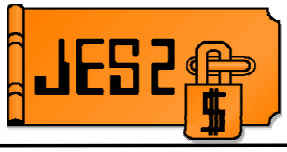


- **SAPI read only access to data sets**
  - RACF JESSPOOL UPDATE access not needed
    - New bit SSS2SRON (on GET) indicates intent to only read data
    - New error SSS2RRON if attempt to modify anything on PUT
  - Support to modify SYSOUT PRIORITY on PUT
    - SSS2RPRI bit and SSS2DPRI field
  - Return max return code/last ABEND code for job (if available)
- **Additional SYSOUT data returned from extended status**
  - Data set level information
  - Other information that requires I/O to retrieve
  - Additional SYSOUT filters

- **Session 2658, JES2 & JES3 z/OS 1.7 SSI , Wednesday 3:00**

There are also enhancements, common to both JES2 and JES3, to the SAPI interface (SSI 79) and extended status (SSI 80). These too will be discussed in more detail in session 2658.

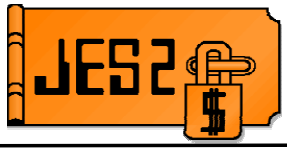
Large (>64K track) SPOOL Data Sets



- **JES2 z/OS 1.7 supports SPOOL data set of up to 1,048,575 (was 64K)**
- **Change the format of a SPOOL address**
  - Currently MTTR (4 bytes)
  - Future MTTtr (still 4 bytes)
    - 8 bits of M, 20 bits of T, and 4 bits of R
- **Implies some changes to format of allocation IOT**
  - TGAE goes from MTT (3 bytes) to MTTTT (5 bytes)
- **Other data areas also affected**
  - $QSE and $HCT sniffer work fields
  - DAS TGM offset and size fields, track limits
- **Cannot be used if >15 records per track**
  - buffer sizes less than 2943 on a 3390
  - **Action: increase spool buffer size**
    - Requires **COLD** start (so schedule well in advance!)

- **Session 2658, JES2 for z/OS 1.7 Large Spool, NJE over TCP/IP, and Other Features, Wednesday 3:00**

Large (>64K track) SPOOL Data Sets *(cont…)*



- **New external, SPOOLDEF LARGEDS=FAIL|ALLOWED|ALWAYS**
  - Applies when a volume is started
  - FAIL causes start to fail if data set is >64K tracks
  - ALLOWED will permit the start and use new MTTtr if >64K tracks
  - ALWAYS will permit all starts and always use new MTTtr
    - ALWAYS intended for testing
- **$T to switch to any values (FAIL to ALLOWED or visa versa)**
- **Once LARGEDS=ALLOWED or ALWAYS CANNOT start z5 or lower**
  - SPOOL data areas incompatible with older releases
- **JES2 will no longer support SPOOLDEF RELADDR=**
  - In particular, RELADDR=NEVER is no longer supported
- **All new volumes will use relative track addressing**
- **Will continue to support absolute track addressing for volumes started on previous releases**
- **Action: use SPOOL read SSI to access SPOOL data directly**

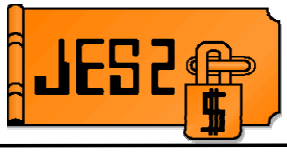
The problem that is being addressed by this line item is the current limit on the size of a SPOOL data set. Before JES2 could address this limit, DFSMS

needed to write support code that allowed a non-extended (non-SMS managed) data set that was greater than 64K tracks. Once that was done, JES2 needed support code to be able to address that much space in a single data set.

JES2 uses 4 byte MTTRs to address records on SPOOL. Using this scheme, we can address up to 64K tracks with 255 records per track. But JES2 formats the tracks with much less than 255 records per track. On a 3390 with the recommended buffer size of 3992 bytes, JES2 used 12 records per track. This implies that we can use some of the bits from the “R” value to supplement the TT value. By borrowing 4 bits, we can get 20 bits or 1M tracks. If the buffer size is too small, such that there are more than 15 records per track, this scheme cannot be used. That is considered a permanent restriction.

The SPOOL read SSI makes these changes transparent to any application that does not look at the contents of the MTTR.

Other goodies



- **$EXIT 49 enhancements**
  - Now called for $SJ command processing and $SJ selection
  - X049IND reflects call type
  - Response byte to bypass duplicate job checking
    - For exits that need to do it themselves
- **Job list command enhancements**
  - OS/390 R4 only implemented filters on commands equivalent to existing commands
  - Almost all keywords now filterable on all commands
  - Many new filters added
    - **BUSY=**, **CRTIME=** on job-level commands (**$*x*J**)
    - **CC=** on output-level commands (**$*x*OJ**)
    - **OUTGRPS**, **RECORDS**, **PAGES** on **$LJ**

Job-level Commands

Command / Filter	$A	$C	$E	$H	$P	$T	$D
CLASS= (C=)	NEW!	NEW!	NEW!	NEW!	NEW!	*	*
PRIORITY= (P=)	NEW!			NEW!	NEW!	*	*
SYSAFF= (S=)	NEW!	NEW!	NEW!	NEW!	NEW!	*	*
HOLD= (H=)		NEW!			NEW!	*	*
SECLABEL=	NEW!	NEW!	NEW!	NEW!	NEW!	NEW!	*
SECLABEL_AFF=	NEW!	NEW!	NEW!	NEW!	NEW!	NEW!	*
SCHENV=	NEW!	NEW!	NEW!	NEW!	NEW!	*	*
SCHENV_AFF=	NEW!	NEW!	NEW!	NEW!	NEW!	NEW!	*
USERID=	NEW!	NEW!	NEW!	NEW!	NEW!	*	*
CARDS=	NEW!	NEW!		NEW!	NEW!	*	*
DELAY=		NEW!			NEW!	*	*
CC=		NEW!			NEW!	*	*
OFFS=	NEW!			NEW!	NEW!	*	*
BUSY=	NEW!	NEW!		NEW!	NEW!	NEW!	NEW!

* - Available prior to z/OS 1.7

Here are all of the job-level filters that were added.

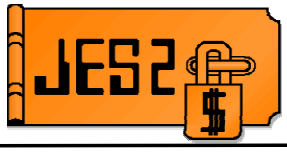
Output-level Commands

Command / Filter	$C O	$P O	$O	$T O	$D O
BURST=	NEW!	NEW!	NEW!	*	*
FCB=	NEW!	NEW!	NEW!	*	*
FLASH=	NEW!	NEW!	NEW!	*	*
FORMS=	NEW!	NEW!	NEW!	*	*
OUTDISP=	NEW!	NEW!		*	*
HOLDRC=	NEW!	NEW!		*	*
PRIORITY=	NEW!	NEW!	NEW!	*	*
PRMODE=	NEW!	NEW!	NEW!	*	*
UCS=	NEW!	NEW!	NEW!	*	*
WRITER=	NEW!	NEW!	NEW!	*	*
RECORDS=	NEW!	NEW!	*	*	*
CC=	NEW!	NEW!	NEW!	NEW!	NEW!
SECLABEL=	NEW!	NEW!	NEW!	*	*
USERID=	NEW!	NEW!	NEW!	*	*

* - Available prior to z/OS 1.7

Here are all of the output-level filters that were added.

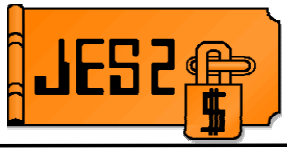
Other goodies



- **Offload device creation time**
  - **Problem:** Spool reload always assigned new creation times for jobs and output
    - Resets counter for purging based on age
  - **Solution:** **OFFLOADn CRTIME=RESET** or **RESTORE**
    - **RESET** – assigns new creation time
    - **RESTORE** – uses original creation time
- **Reachable nodes display**
  - **$DLINE(*nnnn*),NODES**
    - Displays list of all nodes reachable via the line
  - **$DLINE(*nnnn*),NODE=*name***
    - Displays all lines by which node is reachable

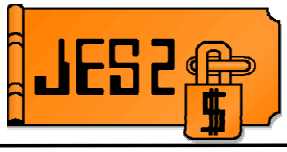
The NODEs subparameter is added to the LINE, L.JT, and L.ST display commands. This can be used in conjunction with the $DNODE, $DPATH, and $D CONNECT parameters to determine information about paths to a node, or to determine specifically why a particular transmitter is not selecting work for a specific node. The NODES parameter must be requested specifically to display as many configurations would result in a very large display.

Drop of Support for Release 4 mode



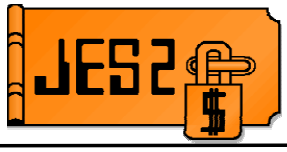
- **JES2 z/OS 1.4 and z/OS 1.5 support 2 modes of checkpoint operation**
  - R4 mode to be compatible with pre z/OS 1.2 members
  - z2 mode to support new features added in z/OS 1.2
- **$ACTIVATE is used to switch between 2 modes**
- **“N-3” release is z/OS 1.2**
  - All compatible releases support z2 mode
    - Note: z/OS 1.2 is no longer supported
  - All “_R4” fields have been deleted
- **JES2 z/OS 1.7 requires MAS to be in z2 mode to migrate to or coexist**
- **$ACTIVATE command deleted in z/OS 1.7 (again)**
  - Don’t worry, it will be back!
- **Action: Issue $ACTIVATE,LEVEL=z2 prior to migration**
- Session 2656, JES2 Migration, Wednesday 1:30

Requirements addressed in z/OS 1.7



- **zBLC**
  - **PUSC1002-464** - JES2 and JES3 NJE support over native TCP/IP
- **SHARE**
  - **SOJES293002** – Delay the $HASP549 message until NJT receive
  - **SOJES292004** – Provide complete control of NJE store-and-forward
  - **SSJES300353** – Enhance the JES SSI 80 extended status interface
  - **SSJES2038887** – Ability to alter priority of output processed by SAPI
  - **SSJES2038885** – JES2 NJE support over native TCP/IP
- **Other requirements**
  - **MR00069427** – Retain original SYSOUT creation date after offload/reload
  - **MR00069758** – NJE over TCP/IP
  - **MR00074624** - $DJOBQ,/Q=XEQ
  - **MR00075651** - $HASP540
  - **MR00076705** – Adding the BFSZ value in msg HASP200 to SMF record type 55
  - **MR050300486** – NJE does not have TCP/IP support
  - **MR0529014658** – JES2 table driven customization constraints for OEM vendors
  - **MR081600245** - $HASP890 to include total record count for job
  - **MR0829022830** – JES2 job transmitter recovery

Requirements addressed in z/OS 1.7



- **Other requirements**
  - **MR083002610** – JES2 should fully utilize SPOOL volumes with >64K tracks
  - **MR1117035645** – Define a JES2 and JES3 job class in which the job log is purged based on return code
  - **MR1128012551** – SDSF with WLM
  - **MR122701390** – JES2 command $CJ (or $PJ) does not offer the DUP option in order to cancel duplicate jobs in the queue
  - **MR0122036132** – Ability to increase NJEDEF NODENUM without all-member warm start
  - **MR05300022428** - $HASP549 records field is too short
  - **MR053003510** - $HASP375 may show insufficient or misleading information
  - **MR0620025414** – Dynamic NODENUM change for JES2
  - **MR0707035020** – JES2 spool support for greater than 64k tracks, ie, mod27
  - **MR0708031946** – JES2 support for spool data sets larger than 64K tracks
  - **MR072203183** - $CO, $DO, $PO, $TO should reject when ALL used with other filtering options
  - **MR0908026157** – Resource shortages on one node cause NJE transmitters to drain on other nodes
  - **MR100203499** – JES2 does not have a command which shows the count of output groups a job has generated
  - **MR1217046210** – JES2 SYMREC improvements