What Are All These JES2 NJE Options? (The A-Zs of NJE)

Tom Wasik
IBM Rochester, MN

Thursday 9:30AM
Session Number 11757
What is NJE?

- Network Job Entry
- Protocol used to send “Jobs” and “SYSOUT” to other nodes
- Protocol – Agreed upon convention for exchanging data
  - Supports multiple operating systems (heterogeneous nodes)
  - Sits on top of a transport layer (BSC, SNA, TCP/IP)
- Jobs – Pre-execution control language (JCL in z/OS)
- SYSOUT – Post execution job output
- Node – a destination to send jobs or SYSOUT
  - JES2 MAS, JES3 complex, VM image
  - NOT necessarily a SYSPLEX, z/OS image, TCP/IP address
Defining NJE

- Define local operating properties
  - NJEDEF statement (init statement or operator command)
- Define your nodes
  - NODE(x) statement (init statement or operator command)
- Define network topology
  - NODE(x) statement
  - CONNECT statements
- Define mapping of nodes to protocol specific structures
  - APPL(x) statements for SNA
  - SOCKET(x) statements for TCP/IP
- Define devices to manage connections
  - Line(x) statements and subdevices
Defining NJE Operating Constants and Limits

- Number of NJE nodes available
  - NJEDEF NODENUM=
- Local node name/number
  - NJEDEF OWNNODE=
- NJE header pool (for BSC and SNA)
  - NJEDEF HDRBUF=
- VTAM resources (note shared with RJE)
  - VTAM buffers - TPDEF SNABUF=
  - VTAM session (connection) - TPDEF SESSIONS=
- BSC buffers
  - TPDEF BSCBUF=
- TCP/IP buffers, etc. are managed by the NETSERV address space
  - No externals to control them
Defining NJE lines

- A line is an abstraction for controlling an NJE connection
  - BSC lines represent real hardware (CTC in modern world)
  - SNA and TCP/IP lines function as abstract BSC lines
    - No real hardware, just a data area to control a connection
- Lines are associated with a transport protocol
  - UNIT=nnnn (for BSC) or SNA or TCP
- BSC and SNA lines can be used for RJE
  - Specifying NJE parameters limits them to NJE
- BSC lines include hardware specifications
Defining NJE lines

- Lines can be associated with an NJE node (NODE=)
  - Used to select line and node when connecting
- Lines have a resistance associated with them
  - Used in network topology calculations
- TRACEIO= controls general tracing
  - LOG= triggers I/O error message for successful I/O (BSC/SNA)
- START= defines initial state of line (when JES2 starts)
- Passwords are only for RJE connections
Line Sub-devices

- NJE protocols define sub-devices for each connection
  - A single connection can be shared by up to 8 pairs of sub-devices
  - Sub-devices are pre-defined as either for jobs or SYSOUT
    - Defined before making the connection
- Each line then specifies the number of transmitters and receivers
  - SYSOUT transmitters (STNUM=)
  - SYSOUT receivers (SRNUM=)
  - JOB transmitters (JTNUM=)
  - JOB receivers (JRNUNM=)
- Defaults come from NJEDEF
Line Sub-devices

- Each transmitter must be paired with a receiver
  - Unpaired sub-devices are not used
- Sub-devices allow balancing of SPOOL and network I/O
  - Prevents line going idle waiting for SPOOL I/O
  - Can slow down individual item transmission time
- Also prevents large items from monopolizing connection
  - Work selection limits on transmitters can create “express lanes”
Line Sub-devices

- When line specifies xxNUM=, sub-devices are dedicated
- Otherwise comes from a pool specified on NJEDEF
  - Pool size defined by LINENUM=
  - If pool exhausted, then no additional NJE connections allowed
    - Unless connecting line has dedicated sub-devices
- Can start and drain individual sub-devices as needed
  - Drain transmitters rather than receivers
    - When receiver drains paired transmitter drains, but not visa versa
    - Restarting transmitter in this case gets traffic moving again
Line Sub-devices

NODE 7

LINE1

L1.JT1
L1.JR1
L1.ST1
L1.ST2
L1.SR1

NODE 4

LINE3

L3.JR1
L3.JT1
L3.SR1
L3.ST1
L3.ST2
Line Sub-devices

• Which is better for transmitter work selection?
  L1.ST1 LIMIT=(0-*)  - No limit
  L1.ST2 LIMIT=(0-5000)  - Express lane (small SYSOUT)

Or

  L1.ST1 LIMIT=(0-5000)  - Express lane (small SYSOUT)
  L1.ST2 LIMIT=(0-*)  - No limit

• First is better in the case that other node has only on SYSOUT receiver
Line and Sub-device commands

- Lines can be started via $S command
  - Starting makes them ready to use
  - Can be automatically started as part of $SN command
- Lines can be reset via $E command
  - Drops the NJE connection
  - Line stays active ($S’ed)
- Lines can be drained (stopped) via $P command
  - Sets line to go inactive when connection drops
  - If not connection, line goes inactive
- Sub-devices can be individually stopped and started
  - $P L1.SR1 drains the SR1 sub-device of line 1
  - $S L2.JT3 starts the JT3 sub-device of line 2
    - Note transmitters are drained if there is no corresponding receiver
Defining Nodes – NODE(n) NAME=

- Each NJE node has an 8 character name and a number
- Internally, output is routed to the node number not the name
  - Changing a node’s name does not affect data routed to it
- Nodes are maintained in the checkpoint to ensure consistency
  - Init deck overrides CKPT specifications
- Must define a node and number for the local node
  - Local node does not have to be 1
  - Requires a COLD start to change local node number
  - Local node name can be changed by operator command
    $T NODE(old_local),NAME=new_local
    - Note old_local must be the local node name not the number
  - Be aware that XCFGROUP= defaults to local node name
Defining Nodes Security – NODE(x)

- **AUTH=** controls what commands are processed (locally) from the node
  - **NET=** controls commands that affect the operation of the target node
  - **DEVICE=** controls command that affect devices associated with the node (line and sub-devices)
  - **JOB=** controls commands that affect JOBs and SYSOUT from or destined to the node
  - **SYSTEM=** If NET=YES, controls ability to affect system functions at the target node
- **AUTH=** cannot prevent display commands for jobs and SYSOUT destined to or originating from the node
Defining Nodes Security – NODE(x)

- Securing the NJE connection (ensuring node is who they say they are)
  - Password= Connection password to secure connection
    - Traditional way to secure a connection
    - Passwords in clear in INIT deck and transmission
  - SIGNON=SECURE|COMPAT Uses encrypted passwords
    - Passwords stored in security product
    - No clear passwords used
- Securing job passwords in NJE streams
  - PENCRIPT= encrypts passwords in headers (1 job card case)
  - NODES class RACF profiles limits need for passwords at all
- Limiting what is sent to or received from a node
  - RECeive= None, jobs, SYSOUT, both
  - TRAnsmit= None, jobs, SYSOUT, both
Defining APPLs and SOCKETS

- APPLs and SOCKETS associate a NJE node with a network construct
  - Node name or number is specified as NODE=
  - APPLs associate a node with a VTAM APPLID
  - SOCKETS associate an NJE node with a IP name/address
- For inbound connections can be created dynamically
  - NJE knows the inbound network address and NJE node
- Can also be used to specify other connection properties
  - What network server to used for outbound connections
    - APPL LOGON=
    - SOCKET NETSRV=
  - APPL LOGMODE= specifies VTAM logmode to use
  - SOCKET SECURE= indicates that TLS is to be used
Associating JES2 with the Network LOGONs and NETSERVs

• LOGON(x) controls the JES2 association with VTAM
  • Defines the APPLID for connections
  • Password for the VTAM APPLID
  • Each LOGON is a JES2 subtask that talks to VTAM
  • Must be started before it can be used ($S LOGON)
  • Other commands include reset ($E) and drain ($P)

• NETSERV(x) controls the JES2 association with TCP/IP
  • Specifies SOCKET used to get IP address/name, PORT, etc
  • Specifies the TCP/IP stack to use
  • Each NETSERV is an address space that talks to TCP/IP
  • Must be started before it can be used ($S NETSERV)
  • Other commands include reset ($E) and drain ($P)
 Associating Lines with Nodes or Connections

- SNA and TCP/IP lines assigned “at random”
  - Can be specified locally when connection started
    $SN,LINE15,A=TESTJES
  - But this is cumbersome
  - Not an issue for BSC lines (physical unit)
- Lines can be associated with APPLs or SOCKETS
  - LINE= keyword on APPL or SOCKET
  - Often referred to a dedicated line
  - Used for outbound and inbound connections
  - Automatically started as part of connection process
- Lines can be associated with a NODE
  - Used when a $SN,N= is issued
  - Used if corresponding APPL has no LINE=
- Reset LINE= value by specifying LINE=0
NJE Topology (How to get from A to B)

- NJE routing is on top of transport routing
  - Transport moves data packets from point to point
  - NJE moves jobs from node to node
    - For BSC transport routing same as NJE routing
- NJE supports concept of Store and Forward (S&F)
  - Intermediate node going from A to B
  - NJE jobs or SYSOUT is fully received on S&F node
  - It is then transmitted to next node on path to destination

Nodes in path from A to B

A → X → Y → Z → B
NJ E Topology (How to get from A to B)

• To store and forward or not to store and forward?
  • Could have transport level connections to all NJE node
    • Store and forward not needed
    • Less overall overhead
• So why use store and forward?
  • Less connections to manage (thus less complex)
  • Can create “gateway” nodes
    • Nodes that connect large groups of data centers
    • Nodes that connect different networks
      • Company A connecting to Company B
      • May want to limit direct connections between companies
    • Gateways are places to add additional security
      • Do not allow company B so send jobs to company A
• When partial network outage, NJE objects can get closer to their destination
• May not have a consistent protocol to get from A to B
NJE Topology (How to get from A to B)

- JES2 supports dynamic routing using NJE path manager records
  - Additional control records exchanged between JES2 nodes
  - All nodes/members know what NJE connections are up
  - When a connection goes down, NJE objects are rerouted around the outage if possible
  - JES2 only NJE partner that used the path manager
- Should I use path manager connections?
  - Note it is not an all or nothing proposition
  - Are most nodes JES2? No, then limited function
  - May want to use within a data center/city but not between cities
  - May not want to use for intercompany gateways
  - Can create surprise routing of NJE objects
NJE Topology (How to get from A to B)

- Defining connection type
  - NODE(x) PATHMGR=YES|NO
    - Defined if connection can generate a connection record
    - Also controls if connection records are sent over connection
    - Must be NO if this not a JES2 node
    - If this is a JES2 node, then both sides must agree
      - *On NODE(A) specify NODE(B) PATHMGR=NO*
      - *On NODE(B) specify NODE(A) PATHMGR=NO*

- Need to define a CONNECT statement for
  - Non-path manager non-adjacent connections
  - Connections beyond a non-path manager connection

Complete your sessions evaluation online at SHARE.org/AnaheimEval

SHARE 2012
NJE Topology (How to get from A to B)

- CONNECT statements is alternative to path manager dynamic routing
  - Again not an all or nothing thing
- CONNECT specify a NODEA= and NODEB= that are assumed connected
  - Can also override the PATHMGR state of connection
- Path manager creates dynamic connections
- Operator/init statements create static connections
- Connect statements are used to determine how to get from A to B
NJE Topology (How to get from A to B)

- Connect statements are used by the NJE path manager to determine how to get from A to B
  - Explores CONNECT statements to determine best path
    - Both static and dynamic connect statements
  - Can maintain from 1 to 8 paths to a node
    - Most installation use 1 path
  - Best path is determined by lower resistance
    - Resistance is installation controlled value for links (lines) and through nodes
    - Generally resistance reflects line capacity/speed and node willingness to do S&F
    - Least hops acts as a tie breaker
NJE Topology (How to get from A to B)

- Use the $D \text{PATH}(\text{node})$ to display the path to a node
- JES2 can maintain from 1 to 8 paths to a node
  - Most installation use 1 path
  - Multiple paths are all considered equal
- PATH process creates a nodes reachable table for a line
  - Lines select from objects destined to any node in its reachable table
- Multiple paths is not the same a multiple lines to a node
  - You can have an unlimited number of lines connecting 2 nodes
  - Lines can use different protocols (SNA, BSC, or TCP/IP)
  - All lines have same reachable nodes table
  - Lines can be on different members
NJE Topology Influencing Paths

• SUBNETs can divide world of notes
  • SUBNET= keyword on NODE statement
  • $DSUBNET(x)$ displays nodes in a subnet
  • Node can be in only one subnet
• Within a SUBNET (among members of a SUBNET)
  • NJE objects for nodes in the subnet are never sent out of the subnet
    • Paths to node in local SUBNET never leave the SUBNET
• From outside a subnet (Looking into a SUBNET)
  • If you can reach one node in a subnet, it is assumed you can reach all nodes in the subnet
• Network topology records about connections between nodes in a SUBNET are not sent out of the SUBNET
NJE Topology Influencing Paths

- Example subnets
  - US and THEM
    - The data center is the US subnet and other nodes is the THEM subnet
    - Assume one gateway out of the data center or SYSPLEX
    - All NJE going outside the subnet is sent out the gateway node
  - Company subnets
    - Different subnets for each company you interact with
    - IBM subnet, LENOVO subnet, RICOH subnet
    - Different gateways to access nodes in each subnet
- Major advantage is you do not need to understand internals of subnet
- Nodes can be deleted and aliased at entry to SUBNET
  - DESTIDs reroute object destined for old node name to new
    - DESTID(KINGSTON) DEST=POK on entry gateway to IBM node
    - LENOVO and RICOH does not need to know KINGSTON node no longer exists
NJE Topology Influencing Paths

- DIRECT=Y|N – the only path to this node is direct path
  - Never use store and forward for objects to this node
  - Only path is via a connected line to this node
- ENDNODE=Y|N – This node does not do store and forward
  - A path processing ends at this node
  - Uses for small nodes (UNIX box? Non-JES2 node?)
- PRIVATE=Y|N – Do not send any connection records out about this node
  - Node not generally known in the network
  - Connection to node not public (only for my member use)
  - Can use alternate (non-direct path) to get objects to this node
NJE Topology
Influencing Paths

- Be careful not to create conflicting network paths
  - Each member builds their own view of the network
  - Inconsistent use of path influencing techniques can cause network LOOPS

On node A
NODE(A) SUBNET= NODE(B) SUBNET=TEST
NODE(C) SUBNET=TEST

On node B
NODE(A) SUBNET=THEM
NODE(B) SUBNET=
NODE(C) SUBNET=THEM

- Connect node A and node B
- Create SYSOUT for node C
- Where does SYSOUT go?
NJE Topology Influencing Paths

- NJEDEF MAXHOP= prevents infinite looping
  - Every node a NJE object passes through increments a hop count
  - If hop count exceeds MAXHOP= the object is held
- Setting it too low can get work stuck in network
- Setting too high can have loop consume much resource
- Setting to 0 disables function
  - Not recommended to set to 0
NJE Topology
Resistance (It is not futile)

- Resistance can be specified on NODEs, LINEs, APPLs and SOCKETs
  - Resistance is relative desire to use a connection
    - Could be based on line speed or node size
  - Connection resistance is calculated by combining various REST= parameters
- Most networks have physical loops
  - A connects to B connects to C connects to A
  - Not the same as a pathing loop
    - A sends to B to get to C … B sends to A to get to C
  - Resistance prevents jobs from going the wrong way
    - Sending job from LA to San Jose via NYC
NJE Topology
Resistance (It is not futile)

- NJEDEF RESTMAX= maximum total path resistance
  - If path resistance to a node is greater than max, path is truncated
  - Prevents sending files wrong way around the world
  - Total path resistance displayed on $D$ PATH
    $$\text{STATUS=} \text{(THROUGH LNE36), REST=} 20, \text{PATH=} \text{(SANJOSE, POK)}$$

- NJEDEF RESTTOL= Used when PATH>1
  - Secondary paths are not explored if resistance of secondary path is more than RESTTOL difference from the primary
  - Prevents long (round the world) secondary path

- NJEDEF RESTNODE= Local node contribution to resistance
  - Combined with LINE, APPL, SOCKET resistance
Starting NJE networking ($SN)

• Use $SN to start an NJE session
  • Start networking to a node - $SN,N=x
    • Uses the NODE parameter to get the line
    • Use line to start networking
  • Start SNA networking - $SN,A=a
    • Sign onto the specified APPLID
  • Start TCP networking - $SN,S=s
    • Sign onto the specified socket
  • Start networking over an NJE line - $SN,LINEx
    • If LINEx is BSC line, then start NJE on that line
    • If not BSC, then NODE= must be specified on the line
    • Processing similar to $SN,LINEx,N=n
Network Resource Monitor

- Goal is to eliminate the need for NJE automation
  - Sets network up at initialization
  - Ensures network stays up in event of errors
- Starts network devices as part of JES2 initialization
  - START=YES in init deck starts device after JES2 starts
  - Similar to START=YES on other devices (default is NO)
  - Works for all starts of JES2
- Applies to LINEs, LOGONs, and NETSERVs
  - Rules for HOT with active NJE over TCP/IP override START=
    - *If connection active, it will stay active over hot start*
Network Resource Monitor Restarting Devices

• \texttt{RESTART=(YES,interval)} controls restarting devices
  • Applies to LINEs, LOGONs, and NETSERVs
  • Starts a device that is drained (due to command or error)
  • Interval is time to wait (in minutes) before attempting start
    • Prevents excessive start attempts
    • Range is 0-1440 minutes (default is 10 minutes)
    • 0 implies use the interval from \texttt{CONNECT=} on NJEDEF
  • If specified in the init deck, device will start post initialization
    • Same as \texttt{START=YES}
  • Can be set via \texttt{$T} \texttt{operator command}
  • Can be set on \texttt{$P} \texttt{command (eg \texttt{$PLINE1,RESTART=NO)\texttt{)}}
  • \texttt{$D} displays time of next restart attempt
    • \texttt{RESTART=(YES,10, 2012.192,10:25)}
CONNECT=(YES, interval) controls restarting connections
- Causes a $SN command to be generated internally
- $SN format depends on where CONNECT specified
- Interval is time to wait (in minutes) before attempting $SN
  - Prevents excessive start attempts
  - Range is 0-1440 minutes (default is 10 minutes)
  - 0 implies use the interval from CONNECT= on NJEDEF
- If specified in the init deck, $SN done post initialization
- $T command can update value
- $D displays time of next restart attempt
  - RESTART=(YES, 10, 2012.162, 11:21)
Network Resource Monitor
Starting/Restarting NJE Connections

- NODE(x) CONNECT=(YES, interval)
  - Searches for LINEs, APPLs and SOCKETs for this node
    - LINE NODE=x
    - APPL NODE=x
    - SOCKET NODE=x
    - line that this NODE(x) LINE= points to if any
  - If already active, no action taken
  - If CONNECT=DEFAULT on device then a $SN is issued
- If CONNECT=NO, then no connect at node level attempted
  - However connection at lower level still possible
Network Resource Monitor
Starting/Restarting NJE Connections

- APPL(x) CONNECT=(YES, interval)
  - Ignored if NJEDEF CONNECT=NO specified
  - If APPL not connected, a $SN,A=x is issued
  - Independent of CONNECT setting on NODE
  - CONNECT=DEFAULT associates restart with the NODE

- SOCKET(x) CONNECT=(YES, interval)
  - Ignored if NJEDEF CONNECT=NO specified
  - If SOCKET not connected, a $SN,S=x is issued
  - Independent of CONNECT setting on NODE
  - CONNECT=DEFAULT associates restart with the NODE
Network Resource Monitor
Starting/Restarting NJE Connections

- **LINE(x) CONNECT=(YES,interval)**
  - Ignored if NJEDEF CONNECT=NO specified
  - If LINE not connected, a $SN based on UNIT= is issued
    - UNIT=nnnn – starts BSC line ($SN,LINEx)
    - UNIT=SNA – requires NODE= to be set, selects appl using rules similar to $SN,N=
    - UNIT=TCP – requires NODE= to be set, selects socket using rules similar to $SN,N=
  - Independent of CONNECT setting on NODE
  - CONNECT=DEFAULT associates restart with the NODE
Network Resource Monitor
Global Controls

- **$E NETWORK ($E NET) command**
  - Simulates $E of all active lines, logons, and netsrvs
  - Connections will be reset immediately

- **$P NETWORK ($P NET) command**
  - Simulates $P of all active lines, logons, and netsrvs
  - Devices will drain once connections are reset
  - Sets global flags to stop automatic restarts and connects

- **$Z NETWORK command**
  - Simulates $ENET + $PNET

- **$S NETWORK command**
  - Starts all devices that are restartable immediately, regardless of interval
  - Starts all connections that are restartable immediately, regardless of interval
  - Resets global flags
Questions?

Session Number 11757