z/OS V2.5 IBM Education Assistant

Solution Name: JES2 Policy expansion

Solution Element(s): JES2

Agenda

- Trademarks
- Objectives
- Overview
- Usage & Invocation
- Interactions & Dependencies
- Upgrade & Coexistence Considerations
- Installation & Configuration
- Summary
- Appendix

Trademarks

- See url http://www.ibm.com/legal/copytrade.shtml for a list of trademarks.
- Additional Trademarks:
  - None

Objectives

- In this presentation we will describe V2R5 enhancements to JES2 policy function that was originally introduced in V2R4.

Overview

- Who (Audience)
  - z/OS system administrators new to JES2
  - JES3 administrators overseeing JES3 to JES2 conversion
  - JES2 administrators who wish to phase out JES2 exit programs
- What (Solution)
  - Enhance capabilities to customize JES2 processing in a way that does not require:
    - low-level programming of JES2 exit programs (Assembler)
    - knowledge of the details of JES2 internal control structures
    - understanding of the JES2 internal processing logic
    - exit recompile for each new release of JES2
- Wow (Benefit / Value, Need Addressed)
  - Reduce the need for specific JES2 skills
  - Provide release-independent way to customize JES2 processing
  - Improve JES2 reliability by isolating JES2 from bugs in JES2 exit programs

Usage & Invocation

- JES2 policy function was introduced in V2R4
- At a very high level, JES2 policy defines in user-level terms what JES2 must do in certain strategic points in JES2 processing.
- Externally, a JES2 policy definition is a JSON object residing in a human-readable editable z/OS data set.
- After having been created by a system administrator, a JES2 policy definition is imported into JES2 by a JES2 operator command, and the policy becomes available for JES2 processing.
- Note that JES2 policies do not completely replace JES2 exits – the exits are still supported.

Usage & Invocation – new policy types

- Only one policy type was originally shipped in V2R4:
  - Policies of type JobConversion are applied by JES2 at the end of job conversion phase (right before exit 44 is called).
- In V2R5, two new policy types are added:
  - SYSOUTGroup – policies of this type are applied to each SYSOUT data set as it is processed by the JES2 for the purposes of creating output groups (right before JES2 exit 40 is called). SYSOUTGroup policy allows to manage attributes of the individual data sets.
  - PreConversion – policies of this type are applied before z/OS converter is called (there is no JES2 exit associated with this point in processing). Converter is able to provide defaults for certain job attributes that were not explicitly set through the JCL of the job. PreConversion policy allows administrator to manage the defaults based on the installation needs.

Usage & Invocation – new built-in functions

- New built-in functions are added in V2R5. They can be used in any policy type:
  - AuthorityCheck()
    - New function to invoke security check from the policy condition.
    - AuthorityCheck() allows to test if specified user has authority of a specified type to a named resource of a specified class.
  - FindNumber()
    - New function to parse a number from a character string (e.g., accounting string of a job).

Usage & Invocation – AuthorityCheck()

AuthorityCheck(class, resource) AuthorityCheck(class, resource, type) AuthorityCheck(class, resource, type, userid) Result is a numeric value	Performs an authority check – tests if specified **userid** has authority of a specified **type** to the **resource** of a specified **class**. The function returns a numeric result of the authority check: - 0 – user has authority of a specified type - 4 - undecided - 8 – user does not have authority of a specified type - 256 + 4 - data not available (job owner cannot be resolved) - 256 + 8 - invalid parameters - 256 + 12 - SAF failure - 256 + 16 - RACF failure	**class** is a SAF resource class name (character string) **resource** is a SAF resource name (character string) **type** is one of the following character values: - 'READ' - 'UPDATE' - 'CONTROL' - 'ALTER' If omitted, 'READ' is assumed. **userid** is a name of the user for the authority check (character string). If omitted, name of the owner of the current job is used, job structure is available to the current policy type)

Usage & Invocation – FindNumber()

FindNumber(string) FindNumber(string, count) Result is a numeric value	Returns a numeric representation of a number in the character **string**. Optional **count** specifies which number to return (first, second, etc.). If **count** is omitted, the first number is returned. If no number is found in the string, zero is returned. "Number" in this context is a sequence of digits surrounded by any characters. E.g. in a string 'xxx1234,5678yy', the first number is 1234 and the second number is 5678.	**string** – character **count** - numeric

Usage & Invocation – new job attributes

- New job attributes have been added:
  - CompletionCode
  - CreatedLocally
  - InputMember
  - JobSeclabel
  - JobHasFailed
  - JobIsPrivileged
  - JobIsProtected
  - MsgClass
  - UjobCorr
  - WithJob
- These attributes can be referenced by any policy type that has access to necessary job control blocks. The list of attributes available for each policy type is documented in the JES2 Installation Exits manual.

Usage & Invocation – new job attributes (2)

CompletionCode	Returns result of job termination in a format similar to HASP165 message. If completion code was not set, the list is empty. Otherwise, list has one or two elements. The first element is the type of completion and the second element provides the details: - CC - second value is the completion code as “nnnn” - JOBRC - second value is the completion code as “nnnn” - JCL ERROR - CANCELLED - ABEND - second value is “Sxxx” or “Unnnn” - CONVERSION ABEND - SECURITY VIOLATION - CONVERSION ERROR - SYSTEM FAILURE - FLUSHED - END OF MEMORY - RC - second value is the completion code as “nnnn”	character list
CreatedLocally	Tests if job went through the input processing locally.	logical

Usage & Invocation – new job attributes (3)

InputMember	Name of the JES2 member where job was submitted.	character
JobHasFailed	Tests if job has failed. If this attribute is invoked before job has completed it returns false.	logical
JobIsPrivileged	Tests if job was submitted via emergency subsystem.	logical
JobIsProtected	Tests if job's output is protected.	logical
JobType	Returns job type: 'Job', 'STC', 'TSU', 'JobGroup', or 'SYSLog'.	character
JobSeclabel	Security label associated with the job.	character

Usage & Invocation – new job attributes (4)

MsgClass	Message class of the job – specified on the MSGCLASS keyword of JOB JCL statement or set by default. Note: MsgClass can be changed by “ModifyJob” action of PreConversion policy.	character
UjobCorr	User-specified portion of job correlator (UJOBCORR= keyword on JOB statement).	character
WithJob	Name of a job specified on the WITH keyword of SCHEDULE JCL statement. If not specified for the job, WithJob attribute is an empty string (string of length 0). Note: WithJob can be changed by “ModifyJob” action of JobConversion policy.	character

Usage & Invocation – PreConversion policies

- Policies of type PreConversion are applied before z/OS converter is invoked to process JCL of a job.
  - There is no JES2 installation exit at this point in JES2 processing.
- z/OS converter is able to apply default values to job attributes that are not explicitly defined by the JCL of the job. The main purpose of PreConversion policy is to manage these default values.
- Syntax of PreConversion policies is similar to syntax of existing JobConversion policy type – a series of definitions, each containing a condition and a set of actions associated with that condition.

Usage & Invocation – PreConversion policies (2)

- In addition to standard actions, supported for all policy types, the following actions specific for this policy type are supported:
  - **SetDefaults** action allows to assign values to default attributes passed to the converter.
  - **ModifyJob** action allows to modify job attributes. Currently, PreConversion policy can only modify **MsgClass** attribute.
- PreConversion policies also have access to most job level attributes. The list of attributes available is documented in the JES2 Installation Exits manual.
- The table in the following chart lists attributes supported by the PreConversion policy. All attributes can be modified through the SetDefaults action except for MsgClass, that can be modified through the ModifyJob action.

Usage & Invocation – PreConversion policies (3)

AllowSYSSYM	Specifies whether system symbols are substituted in batch jobs.	logical
CommandDISP	Disposition of MVS commands.	character
GenType6	Specifies whether JES2 is to produce type 6 SMF records for job.	logical
GenType26	Specifies whether JES2 is to produce type 26 SMF records for job.	logical
MsgClass	Message class of the job – specified on the MSGCLASS keyword of JOB JCL statement or set by default. Note: MsgClass can be changed by “ModifyJob” action of PreConversion policy.	character
Region	Default region size to be assigned to each job step.	character
ReqACCT	Specifies whether an account number is required.	logical

Usage & Invocation – PreConversion policies (4)

ReqPgmrName	Specifies whether a programmer name is required.	logical
SuppressJobLog	Specifies whether the JES2 job log is to be printed.	logical
SuppressSYSOUT	Specifies whether SYSOUT data is to be written.	logical
SWAAbove	Specifies whether all eligible schedule work area (SWA) control blocks created for this job should be placed above or below 16 megabytes in virtual storage.	logical
TakeIEFUJP	Specifies whether the IEFUJP installation exit is to be taken when this job is purged.	logical
TakeIEFUSO	Specifies whether the IEFUSO installation exit is to be taken when a SYSOUT limit is reached for this job.	logical
UseJournal	Specifies whether job-related information is to be saved in the job's journal.	logical

Usage & Invocation – PreConversion policies (5)

- This is an example of PreConversion policy:

```
{  "policyName":     "PCONV1",
   "policyVersion":  1,
   "policyType":     " PreConversion ",
   "definitions":
      [
        { "condition" : " JobIsProtected  ",
           "actions"  :
              [
                 {  "action"    : " ModifyJob ",
                    "attribute" : " MsgClass ",
                    "value"     : " 'A' "
                 },
                 {  "action"  : " SendMessage ",
                    "message" : " JobName || ' now has MSGCLASS: ' || MsgClass"
                 }
              ]
        }
      ]
}
```

Usage & Invocation – PreConversion policies (6)

- This is an example of PreConversion policy:

```
{  "policyName":     "PCONV2",
   "policyVersion":  1,
   "policyType":     " PreConversion ",
   "definitions":
      [
        { "condition" : " Match(JobType,'Job')  ",
           "actions"  :
              [
                 {  "action"    : " SetDefaults ",
                    "attribute" : " Region ",
                    "value"     : " '1M' "
                 },
                 {  "action"  : " SendMessage ",
                    "message" : " JobName || ' now has default region 1M' "
                 }
              ]
        }
      ]
}
```

Usage & Invocation – SYSOUTGroup policies

- Policies of type SYSOUTGroup are applied to each SYSOUT dataset before the dataset is processed by JES2 for the purposes of creating the output groups.
  - SYSOUTGroup policy corresponds to existing JES2 installation exit 40.
- The purpose of SYSOUTGroup policy is to modify certain attributes of the SYSOUT dataset in order to influence the grouping process.
- Syntax of SYSOUTGroup policies is similar to syntax of existing JobConversion policy type – a series of definitions, each containing a condition and a set of actions associated with that condition.

Usage & Invocation – SYSOUTGroup policies (2)

- In addition to standard actions, supported for all policy types, the following actions specific for this policy type are supported:
  - **DeleteDataset** action allows to delete the dataset
    - Non-SPIN datasets are logically deleted. They physically stay on SPOOL until the job is purged.
    - SPIN datasets are immediately purged from system.
  - **Modify** action allows to modify certain attributes of the dataset.
- The table in the following chart lists attributes supported by the SYSOUTGroup policy. Attributes that can be modified by the policy are flagged as such in the table.

Usage & Invocation – SYSOUTGroup policies (3)

ByteCount	Data set size in bytes.	numeric
DestIsLocal	Tests if data set output destination is local.	logical
DDName	DD (data definition) name.	character
DSClass	Output class of the data set. Note: DSClass can be changed by “Modify” action of SYSOUTGroup policy.	character
DSDest	Output destination of the data set. Note: DSDest can be changed by “Modify” action of SYSOUTGroup policy.	character
DSLRECL	Maximum record length of the data set	numeric

Usage & Invocation – SYSOUTGroup policies (4)

DSName	Returns user-specified dataset name. It was either specified on DD statement or defaults to "?".	character
DSNode	Destination node name for data set.	character
DSPrty	Data set priority. Note: DSPrty can be changed by “Modify” action of SYSOUTGroup policy.	numeric
DSSeclabel	Security label associated with the data set.	character
IsJesDataset	Tests if this is a JES data set.	logical
Owner	User id of the data set creator.	character

Usage & Invocation – SYSOUTGroup policies (5)

PageCount	Data set size in pages.	numeric
ProcStep	Procedure step name that created the data set.	character
RecordCount	Data set size in records.	numeric
StepName	Step name that created the data set.	character

Usage & Invocation – SYSOUTGroup policies (6)

- This is an example of SYSOUTGroup policy:

```
{  "policyName":     "SOG1",
   "policyVersion":  1,
   "policyType":     " SYSOUTGroup ",
   "definitions":
      [
        { "condition" : " isJesDataset  ",
           "actions"  :
              [
                 {  "action"    : " Modify ",
                    "attribute" : " DSDest ",
                    "value"     : " 'NODE3' "
                 },
                 {  "action"  : " SendMessage ",
                    "message" : " DDName || ' sent to ' || DSDest "
                 }
              ]
        }
      ]
}
```

Interactions & Dependencies

- Software Dependencies
  - None
- Hardware Dependencies
  - None
- Exploiters
  - Any installation that has a need to customize JES2 processing.

Upgrade & Coexistence Considerations

- To exploit this solution, all systems in the Sysplex must be at the new z/OS level: No
  - JES2 compatibility APAR OA58722 is required to tolerate new function.
  - New policy types but will not be visible to members at z/OS 2.4 level.
  - If JobConversion policy is created on z/OS 2.5 with new attributes not supported on z/OS 2.4, such policy will be rejected by members at z/OS 2.4 level.
- List any toleration/coexistence APARs/PTFs.
  - JES2 compatibility APAR OA58722
- Compatibility APAR OA58722 is also recommended for fallback to z/OS 2.4 if policies exploiting new features have been created by z/OS 2.5 and remain in the JES2 checkpoint.

Installation & Configuration

- No special installation is required.
- Planning considerations for using JES2 policies are documented in JES2 Installation Exits publication.

Summary

- In this presentation we described V2R5 enhancements to JES2 policy function that was originally introduced in V2R4.

Appendix

- Publications
- z/OS V2R5.0 JES Application Programming – SA32-0987-50
- z/OS V2R5.0 JES2 Commands – SA32-0990-50
- Z/OS V2R5.0 JES2 Diagnosis - GA32-0993-50
- z/OS V2R5.0 JES2 Initialization and Tuning Guide – SA32-0991-50
- z/OS V2R5.0 JES2 Initialization and Tuning Reference – SA32-0992-50
- z/OS V2R5.0 JES2 Installation Exits – SA32-0995-50
- z/OS V2R5.0 JES2 Macros – SA32-0996-50
- z/OS V2R5.0 JES2 Messages – SA32-0989-50
- z/OS V2R5.0 MVS JCL Reference - SA23-1385-50
- z/OS V2R5.0 MVS Using the Subsystem Interface – SA38-0679-50