

UNIVERSITÉ DU MAINFRAME



Tivoli Workload Scheduler End to End « E2E »



Jean-Gabriel WEYER
Jg_weyer@fr.ibm.com

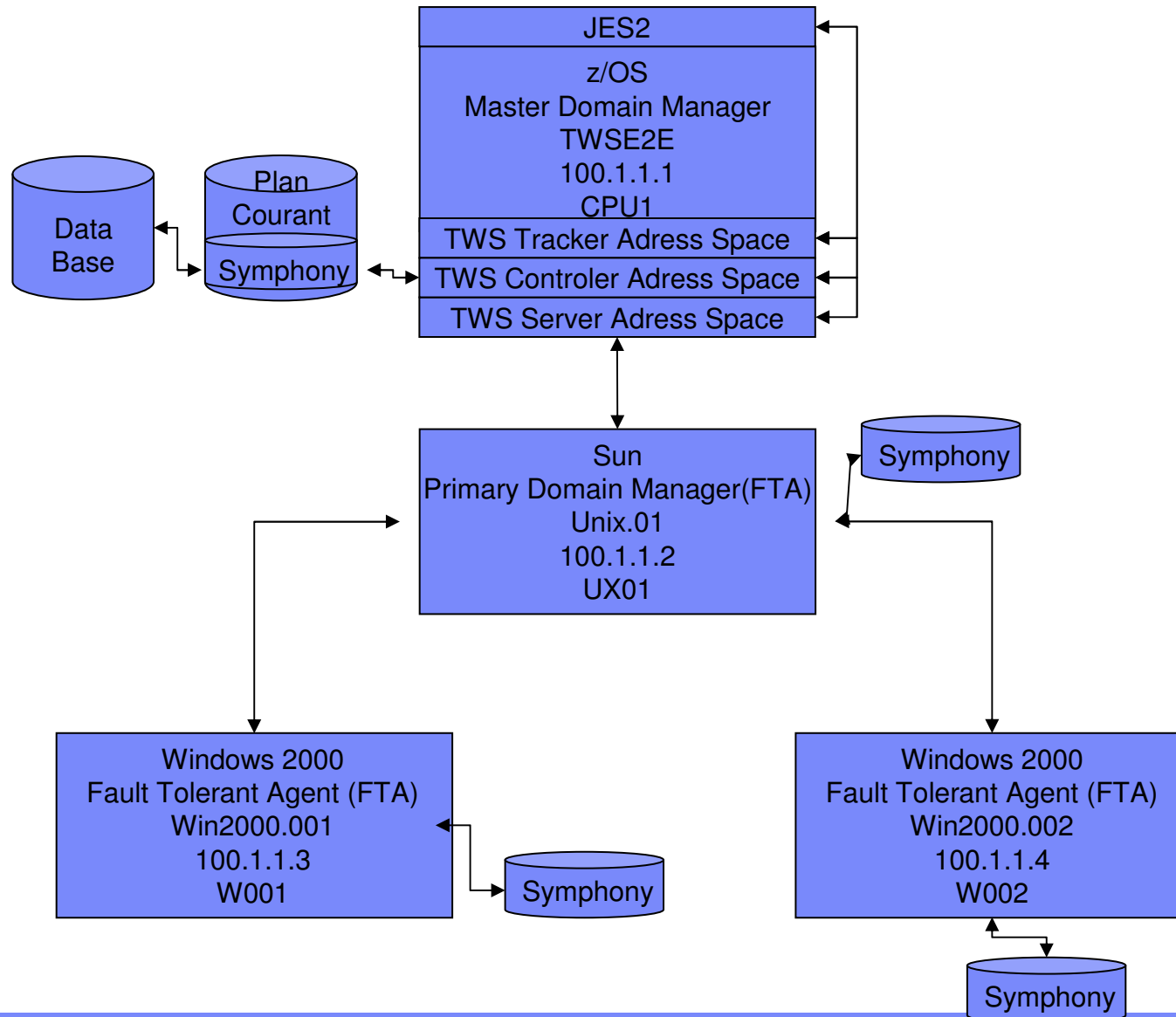
© 2006 IBM Corporation

Agenda

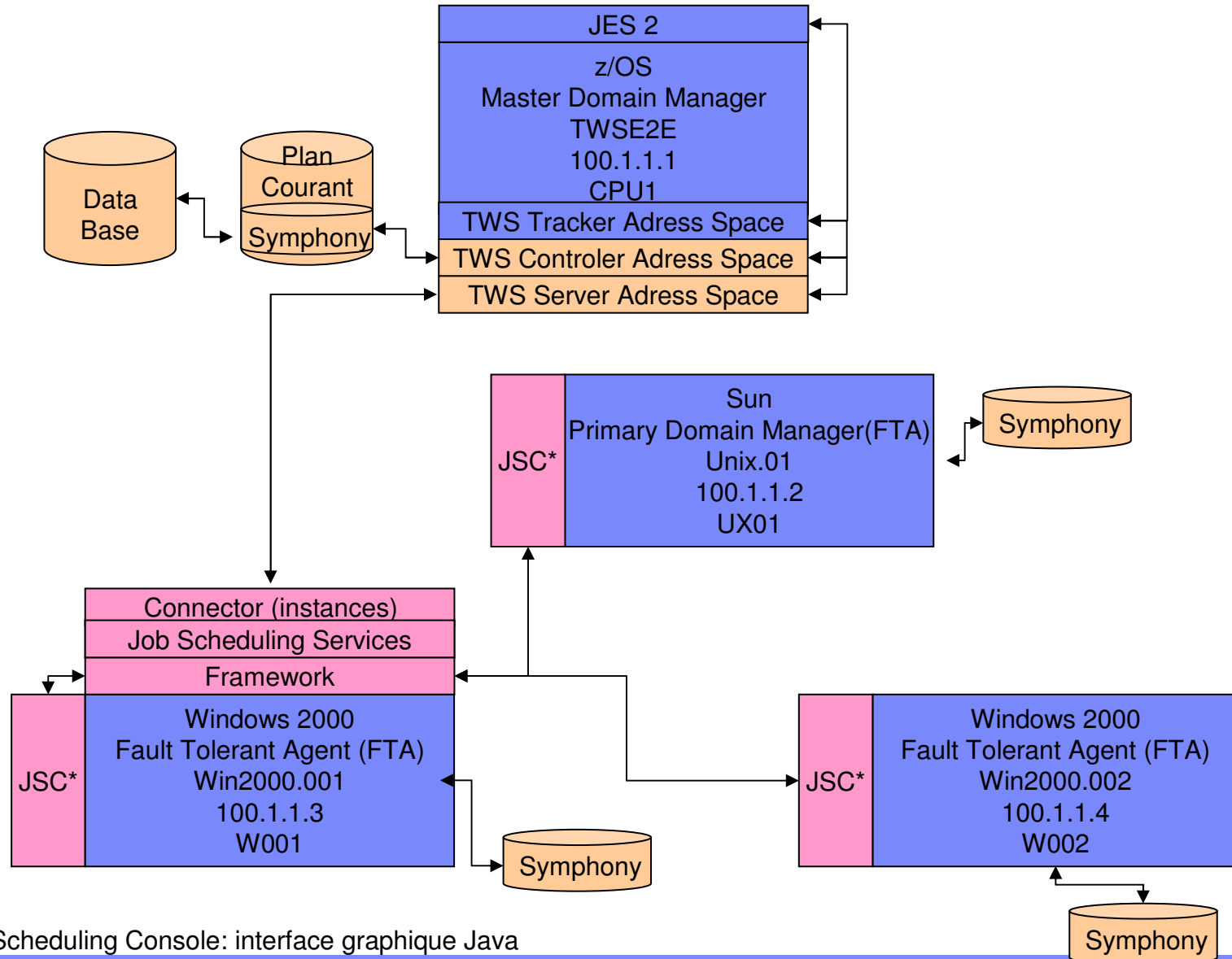
- Architecture « E2E »
- Nouvelle console JSC
- Interface avec WLM
- Gestion centralisée des sysouts de l'environnement distribué
- TWS for Applications
- Questions/réponses



Exemple de configuration « E2E » simple: 1) les agents FTA



Exemple de configuration « E2E » simple: 2) les consoles JSC



* Job Scheduling Console: interface graphique Java

- end to end server parmlib member (SERVOPTS) :

- ▶ member(SERVOPTS) statements:

SERVOPTS TPLGYPRM (**TOPOLOGY**)

*/*member with topology definitions */*

- ▶ member(TOPOLOGY) statements:

TOPOLOGY TPLGYMEM (**TPDOMAIN**)

*/*member with domain + FTA descriptions*/*

- ▶ member(TPDOMAIN) statements:

DOMREC

DOMAIN(Unix.01) */*domain name for 1st domain */*

DOMMNGR(UX01) */* Domain Manager FTA name */*

DOMPARENT(MASTERDM) */* domain parent is MASTERDM */*

CPUREC

CPUNAME(UX01) */* FTA name ("OPC" WS name) */*

CPUDOMAIN(Unix.01) */* the TWS domain name for this FTA */*

CPUREC

CPUNAME(W001) */* FTA name ("OPC" WS name) */*

CPUDOMAIN(Unix.01) */* TWS domain name for this FTA */*

- end to end server parmlib member (SERVOPTS) :
 - ▶ member(SERVOPTS) statements:

```
SERVOPTS
    TPLGYPRM (TOPOLOGY) /*member with topology definitions */
    USERMAP (JSCUSERS) /* member with JSC users mapping to TSO users */
```
 - ▶ member(TOPOLOGY) statements:

```
TOPOLOGY
    USRMEM (TPUSER) /* member with Windows user-pswd defs */
```
 - ▶ member(TPUSER) statements:

```
USRREC
    USRCPU(W001) /* W001 FTA is a Windows System */
    USRNAM(maestro) /* the user name defined for TWS jobs submission */
    USRPSW('maestro') /* password for the previous user name */
```
 - ▶ member(JSCUSERS) statements:

```
USER 'maestro@ Win2000.001-REGION' RACFUSER(TSO00001)
/* 'maestro' JSC user will have 'TSO00001' rights to access TWS DB & plan on z/OS */
/* 'maestro' is here the userid defined in the Tivoli FrameWork */
```

Agenda

- Architecture « E2E »
- Nouvelle console JSC
- Interface avec WLM
- Gestion centralisée des sysouts de l'environnement distribué
- TWS for Applications
- Questions/réponses



Job Scheduling Console 8.3

The screenshot displays the Job Scheduling Console 8.3 interface. A dialog box titled "Define a New Engine for Job Scheduling Console" is open in the foreground, showing settings for a new engine named "MyOwnEngine". The dialog includes sections for Information, Connection Data, and Connection Profile.

The background window, "All Scheduled Job Streams - TWS83", shows a tree view of job streams and a table of job details. The table lists job names, numbers, streams, workstations, and their current status.

Job Name	Job Number	Job Stream	Workstation	Workstation	Status
MAKEPLAN	0	FINAL	OLDDIMEB...	OLDDIMEB...	Error
SWITCHPL...	0	FINAL	OLDDIMEB...	OLDDIMEB...	Waiting
CREATEP...	0	FINAL	OLDDIMEB...	OLDDIMEB...	Waiting
UPDATEST...	0	FINAL	OLDDIMEB...	OLDDIMEB...	Waiting

The dialog box "Define a New Engine for Job Scheduling Console" contains the following fields:

- Information:** Engine Name: MyOwnEngine
- Connection Data:** Engine Type: Distributed; Host Name: localhost; Port Number: 31117; Realm Name: TWSREALM; Remote Server Name: (empty)
- Connection Profile:** User Name: joelblow; Password: (masked); Save Password: (unchecked)

Buttons: OK, Cancel

Tivoli Workload Scheduler 8.3 Java API

- Export APIs for remote job management and monitoring
- TWS 8.3 architecture based on J2EE (Java2 Enterprise Ed.)
- Enterprise Java Bean (EJB) remote interfaces accessible via standard Remote Method Invocation (RMI) connection
- Access to all ITWS functional areas
 - ▶ Distributed ITWS database
 - ▶ Distributed ITWS plan (including submit functions)
 - ▶ z/OS ITWS database
 - ▶ z/OS ITWS Plan (including submit functions)
- Documented and supported - Javadoc provided for both data model and connector

Agenda

- Architecture « E2E »
- Nouvelle console JSC
- Interface avec WLM
- Gestion centralisée des sysouts de l'environnement distribué
- TWS for Applications
- Questions/réponses



TWS z/OS 8.2 - WLM SE integration – overview

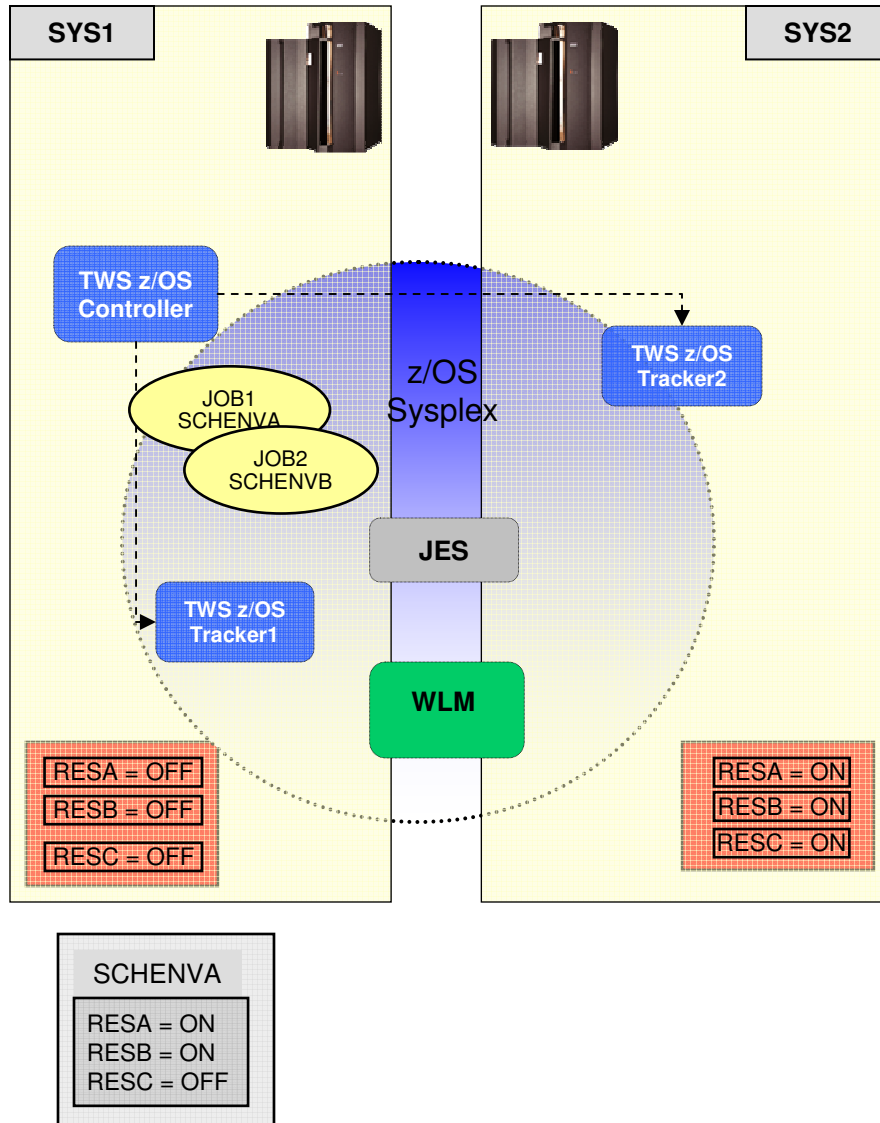
❖ TWSz-WLM integration enhancement, main points

- WLM SE introduced in TWSz definitions (Application Description and Current Plan)
- no more need to manually tailor each JCLs anytime a SE must be associated to a job
- fast detections of SE problems related to TWSz operations
- automatic re-submission of jobs waiting on SE availability
- multiple sysplex configuration supported
- multiple jesplex within sysplex boundaries configuration supported

❖ TWSz-WLM integration enhancement, business value

- load balancing of scheduling workload
- maximizing the usage of system resources
- better service levels handling

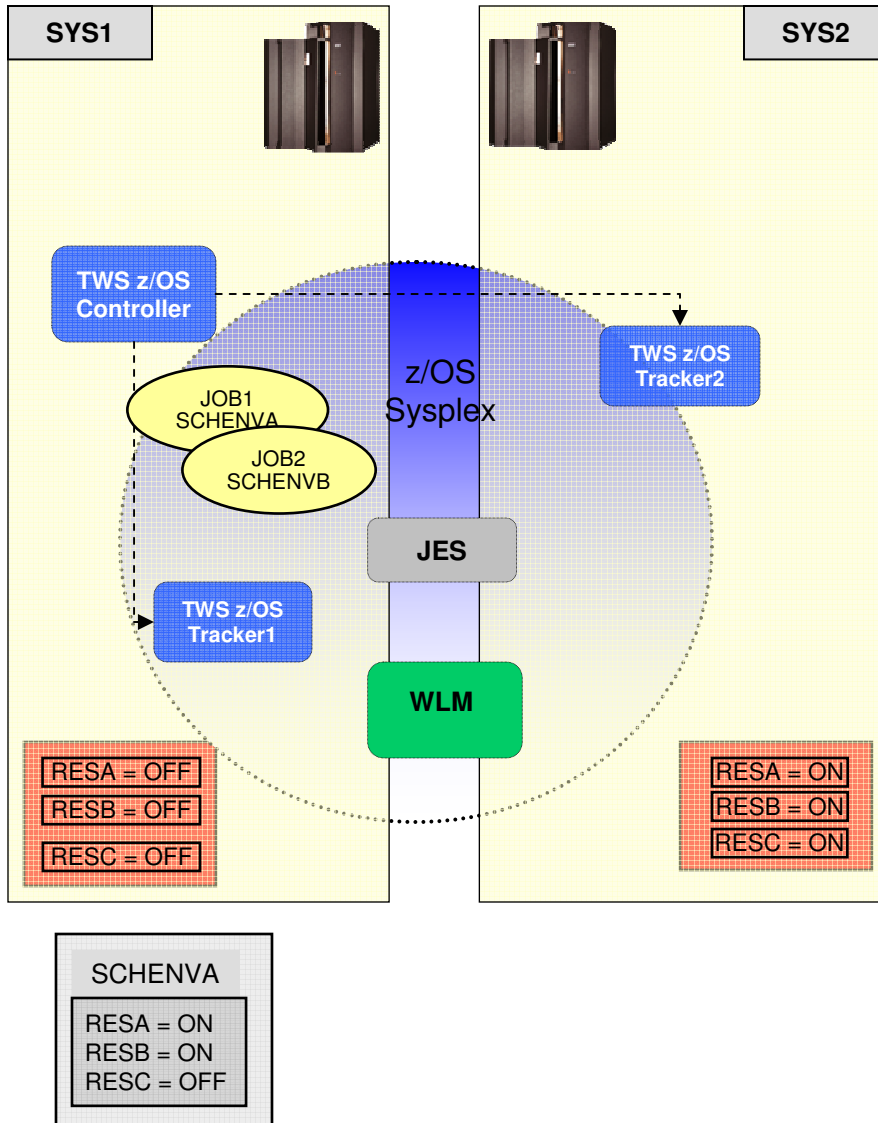
WLM integration : SE definition in TWSz operations



- Associating a WLM Scheduling Environment to TWS for z/OS operations (Application Description and Current Plan)**
- ▶ ISPF - granular
 - ▶ PIF - granular
 - ▶ Mass Update - massive
 - ▶ Batch Loader - massive
 - ▶ EQQUX001 - massive
 - ▶ EQQDPX01 - massive

No more need to edit JCLs to insert SCHENV parameter

WLM integration : SE monitoring in TWSz operations



Monitoring

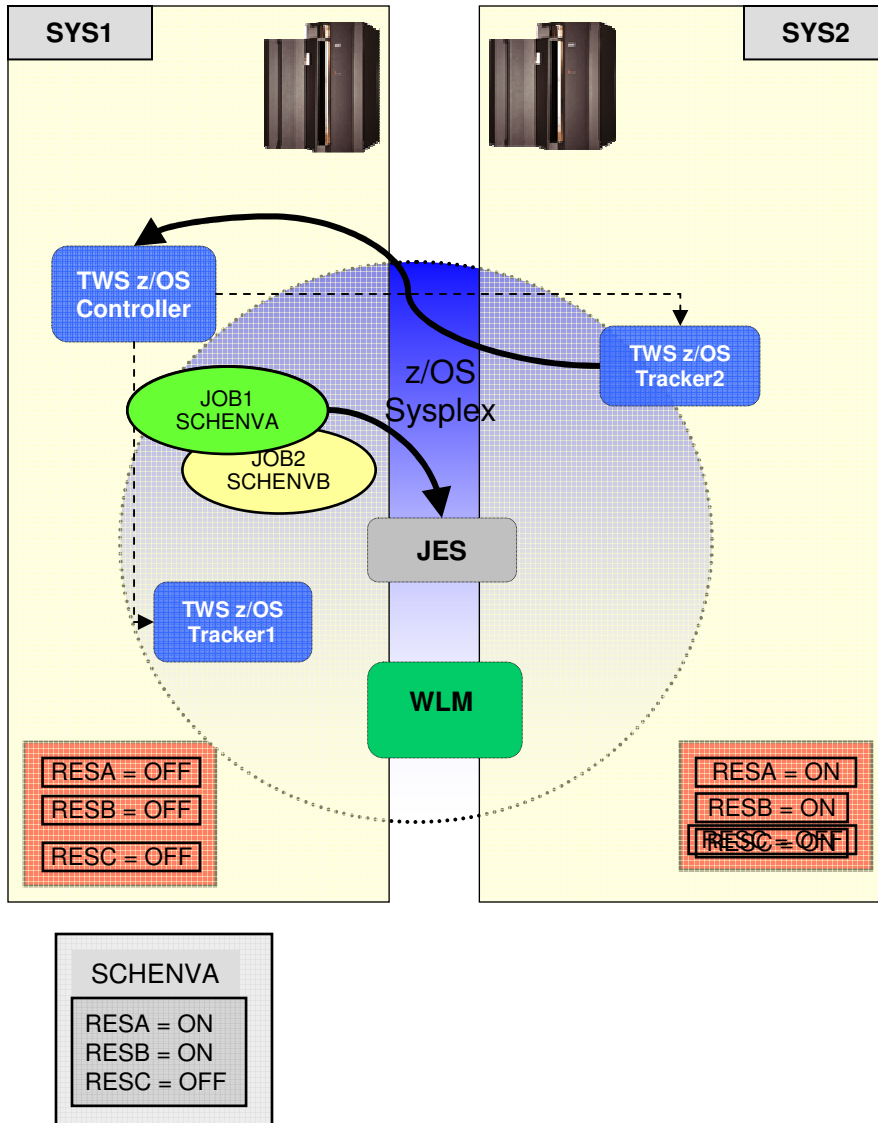
- ▶ Before submitting jobs, TWSz checks for SE status
- ▶ SE not available or not existing prevent job submission
- ▶ Jobs with SE not available are assigned
 - *Waiting for SE* extended status
- ▶ Jobs with SE not existing are assigned
 - *SEUN* error code



Possibility to track SE related problems, via TWSz Dialogs:

- ❖ monitoring on SE
- ❖ monitoring on new extended status

WLM integration : Automatic resubmission of jobs waiting for SE



Automatic resubmission

- ▶ Trackers activate a mechanism listening SEs status
- ▶ It produces a new event as soon as a SE gets available
- ▶ When Controller receives the event, jobs waiting for SE are automatically retried



No manual intervention for jobs holding for SE availability

Agenda

- Architecture « E2E »
- Nouvelle console JSC
- Interface avec WLM
- Gestion centralisée des sysouts de l'environnement distribué
- TWS for Applications
- Questions/réponses

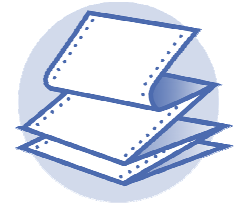


How do I manage all of these outputs ?

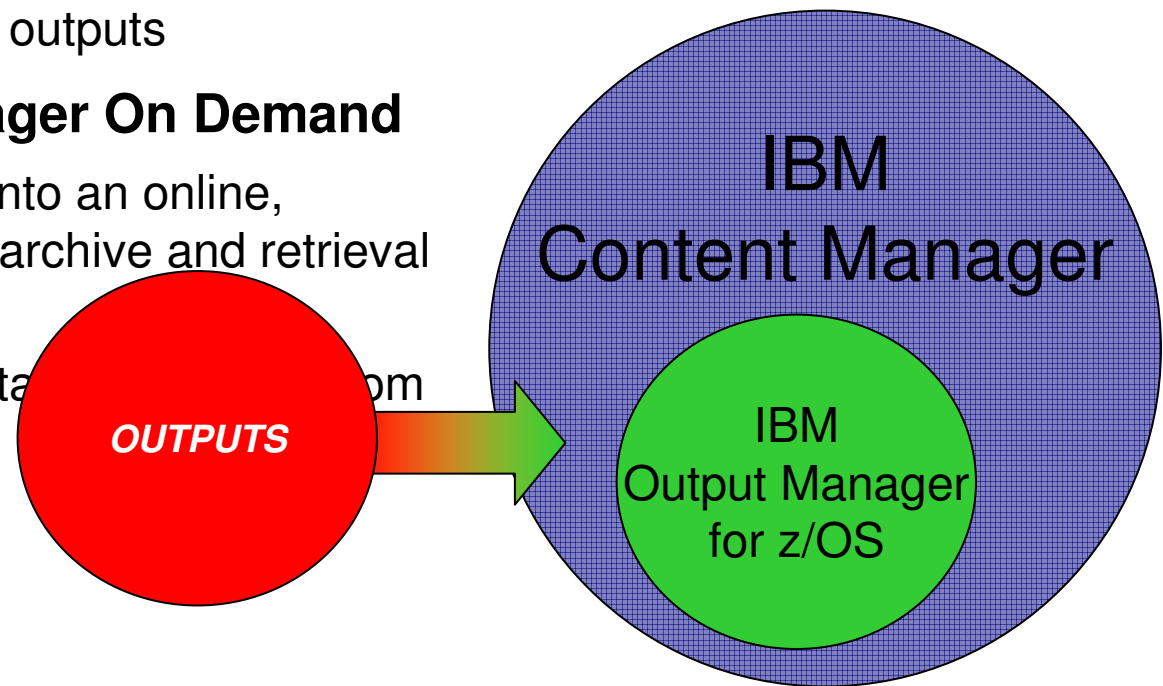
- JCL Job outputs are being stored on datasets or sent to JES output queues to be processed at the later stage
- In the Workload Scheduler E2E configuration, logs and outputs are also produced and stored locally on the distributed server machines where the job actually runs.
 - ▶ Potential risk of data loss
 - ▶ Storage space required for logs on each Agent
- In E2E, viewing logs is done via the JSC interface or ISPF GUI
 - ▶ JSC requires the installation of software rather than a pure browser interface
 - ▶ Advanced job log searching is not possible
- Viewing z/OS job outputs is achieved by using either the JSC, the ISPF GUI, or SDSF (or a similar product).



IBM Output Management Solutions



- **IBM Output Manager for z/OS**
 - Online Viewing and Report Distribution Tool
 - Includes Web Browser Interface and browse facility for third party archiving products
 - Available only for JES outputs
- **IBM DB2 Content Manager On Demand**
 - integration of joblogs into an online, electronic information archive and retrieval system
 - JES outputs, z/OS Data from the distributed servers



What are the benefits within Workload Scheduler

- Single automated and centralized archive for all job log records.
- Easy to use search / retrieve interface via the web
- Elimination of manual activities associated with management of log records distributed around the organisation.

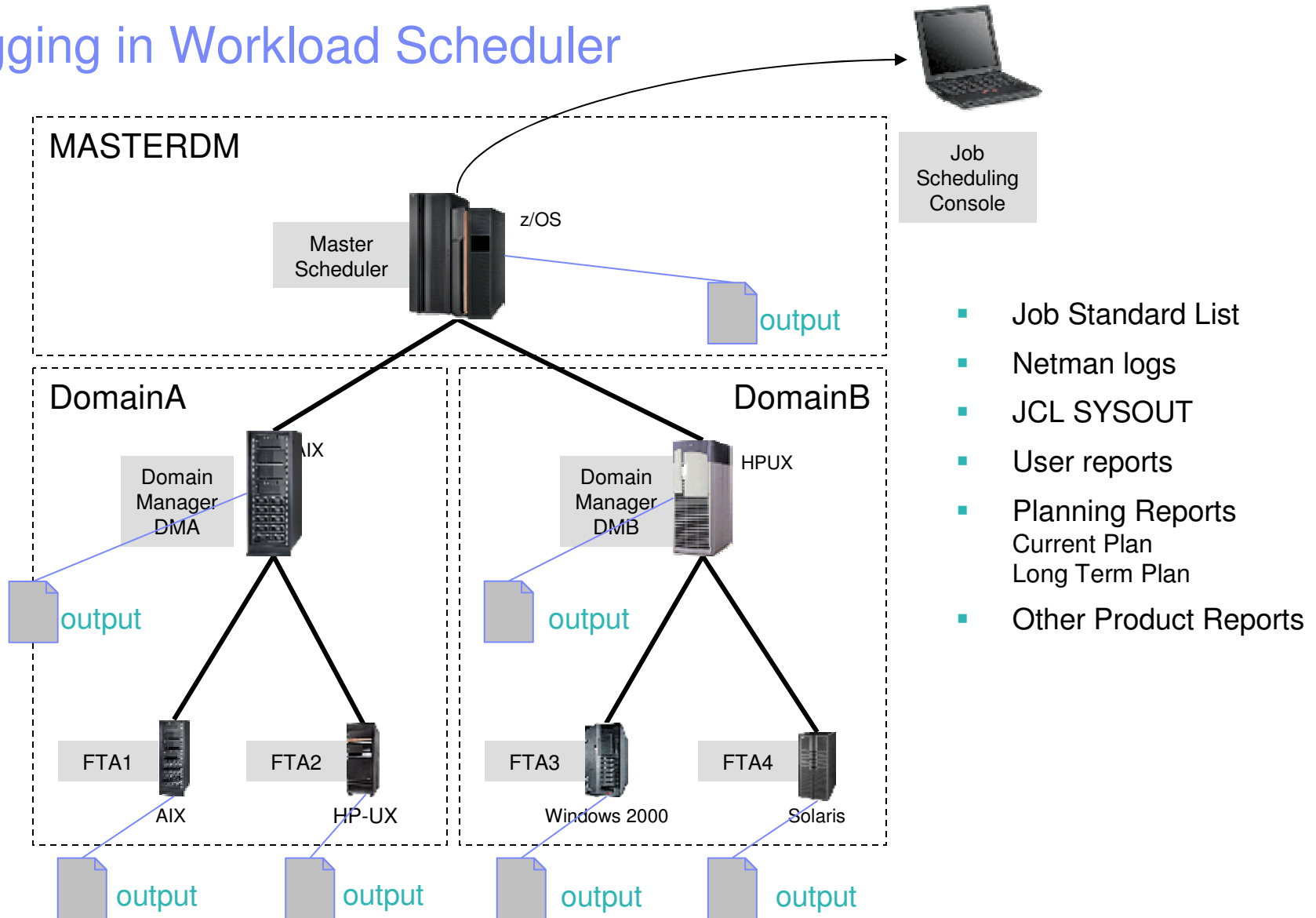


Let's see if there is another solution that provides...

- Multi-Platform and Fault Tolerant capability
- Integration with existing Output Management solution on Mainframe
 - ▶ One single joblog per job
 - ▶ Joblog sent to JES Spool output class where existing archiving product operates
- Header text with relevant job info added to Joblog
- Joblog retrieval executed as soon as job completes (successful or fail)



Logging in Workload Scheduler



- Job Standard List
- Netman logs
- JCL SYSOUT
- User reports
- Planning Reports
Current Plan
Long Term Plan
- Other Product Reports

Why not using EXIT7 ?

A possible solution is to modify EQQUX007 to:

- submit a job any time an operation running on FTA goes to C (Complete) or E (Error) status

The submitted job will:

- use PIF interface to retrieve joblog from FTA
- adopt the same job name as the operation job name that executed on the target FTA as per the TWS for z/OS Plan
- extract necessary information for the header text and copy it, as well as joblog, to a dedicated SYSOUT Class

The existing mainframe archiving product can then archive the header and joblog for the job



**This is one sample solution.
Many customizations are allowed!**

Solution Details

TWS for z/OS plan

Application id	Operation WS no.	Jobname	S
F100DWTESTJOBDEP	DUMY 005	DUMYSTRT	C
F100DWTESTJOBDEP	CPU 010	ADHOC1	C
F100DWTESTJOBDEP	CPU 015	ADHOC2	C
F100DWTESTJOBDEP	F100 020	F100J003	C
F100DWTESTJOBDEP	F100 025	F100J004	C
F100DWTESTJOBDEP	DUMY 255	DUMYEND	C

EQQUX007 exit

Analyse and logic:

- * Is TWS for z/OS calling task the Event Manager (EM)?
- * Yes -> Is it a FTW workstation (Starts with F)?
- * Yes -> Is Status C or E and is it a "job event"?
- * Yes -> Extract job name and other information for header from EQQUX007 parameters
- Submit job with extracted job name

Job submitted by the EQQUX007 exit

1. Job step - Execute REXX program that
 - Retrieves the FTW job log using TWS for z/OS PIF
 - PIF program is called with parameters: ContSubSys, ApplicationID, ApplicationInputArrivalTime, OperationNumber
 - Create header information from information passed from EQQUX007 exit and extracted from the retrieved job log
 - Write header and retrieved job log to temporary and passed dataset
2. Job step - Execute IEBGENER job that
 - Reads header and job log from dataset created in step 1 (allocated to SYSUT1 dd-card)
 - Writes the read data to dedicated JES SYSOUT class (defined on SYSUT2 dd-card)

Note that step 2 will not be executed if return code <> 0 in step 1.

Also note that the header and job log can be passed to step 2 using a physical dataset. If a physical dataset is used to pass header and job log, a step 3 should be added that deletes the dataset if return code = 0 in step 1 and step 2.

Benefits

- Simple: Low Cost of Ownership
 - EQQUX007 and submitted job components only
- Perfectly integrated within TWS for z/OS Controller
 - not necessary to develop programs or to deploy Output Management solution Agents to handle retrieval of joblogs
- Easy to insert logic on existing EXIT7 programs
- Can be combined with any Archiving products on Mainframe
- Uses existing E2E functionalities and topology
- It can be Fault Tolerant
 - if there is no connection with FTW retrieval jobs will end in error
 - retrieval jobs can be manually rerun when connection is re-established

To Consider....

- **Fault Tolerance: retrieval jobs failing due to connection lost with FTW**
 - The process can be monitored and simplified by defining the EQQUX007 submitted job in the TWS for z/OS ETT table
 - Operators or Planners can then be notified of failures and they can rerun the jobs when the connection is re-established
- **Performances**
 - There could be a potential issue when many jobs are being executed and completed in the FTW within a short time interval
 - Concurrent usage of PIF interface by other Applications

Agenda

- Architecture « E2E »
- Nouvelle console JSC
- Interface avec WLM
- Gestion centralisée des sysouts de l'environnement distribué
- TWS for Applications
- Questions/réponses



JSC & XA – Overview

Through the Job Scheduling Console you can:

- ❖ Create/Modify extended agent workstations.
- ❖ Create/Modify Jobs whose aim is to run an extended agent job.
- ❖ Query the list of existing R/3 Standard Jobs, Business Information Warehouse InfoPackages and ProcessChains.
- ❖ Create/Modify R/3 Standard Job properties and step list.
- ❖ Query the list of existing variants for a specific ABAP.
- ❖ Create/Modify variants.
- ❖ Get the whole definition of an existing InfoPackage.

In addition to these standard tasks, the JSC provides a set of specific

R/3 panels that allows you to:

JSC & XA – Standard “Extended Agent” Jobs

For Oracle, PeopleSoft and z/OS Access Methods. Either R/3 Access Method when the JobDef option has been removed from the r3batch.opts file.

Properties - Job Definition Maestro

General

Task

Task Type
Extended Agent Task

Command
-operator OPD -process XRFWIN -type 'SQR Report' -runcontrol 1
Add Parameter...

Return Code Mapping
Expression

OK Cancel

PeopleSoft Example

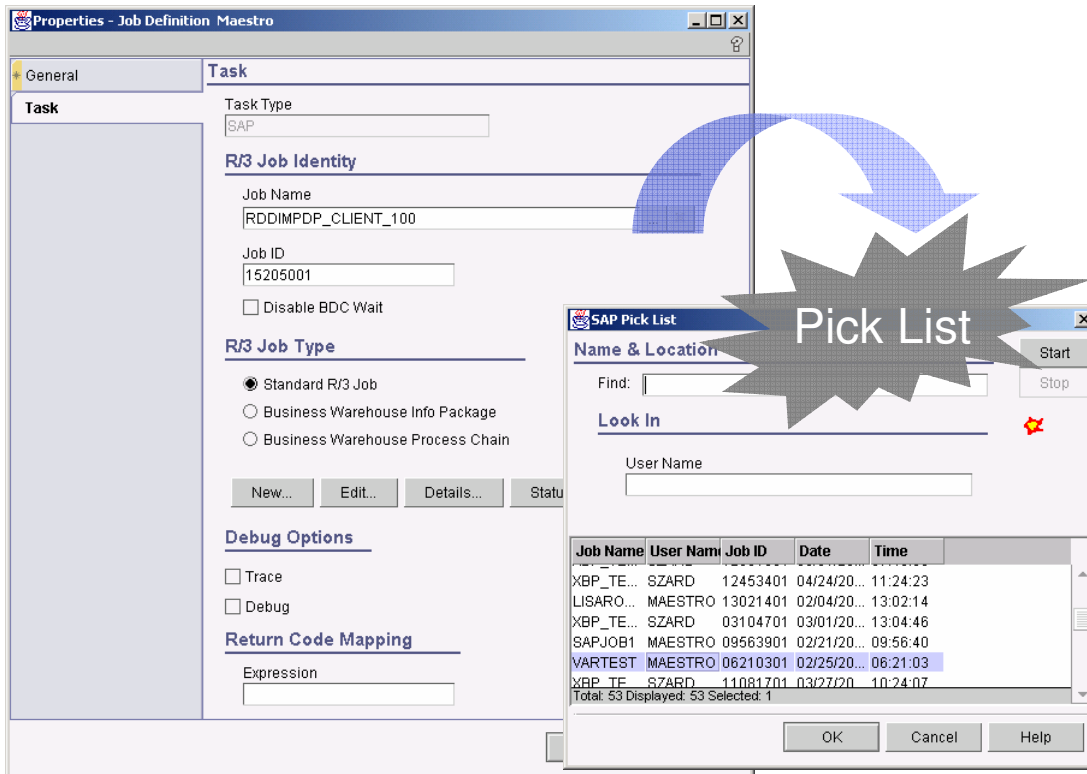
From the JSC you can create standard Extended Agent Jobs.

In the General Tab you should put general information such as the target extended agent cpu, login name and recovery options.

In the Task Tab you need to put the extended agent command, as required by each specific access method.

You could either insert a Return Code Mapping expression.

JSC & XA – R/3 Standard Jobs



From the JSC you can create SAP Jobs.

Actually this is a standard TWS Job whose aim is to run an R/3 Job.

The JSC helps you during this definition phase by providing three types of query for:

- Standard R/3 Jobs
- BIW InfoPackages
- BIW ProcessChains

JSC & XA – R/3 Standard Job Steps

The screenshot shows the 'SAP Job Definition' dialog box with the 'Steps' tab selected. The 'Steps' list on the left contains one step, 'RFBILA00'. The configuration fields for this step are as follows:

- Type: ABAP
- Name: RFBILA00
- User: SZARD
- Var/Param: TEST
- Target Host: (empty)

Print Parameters:

- Number of Copies: 001
- List Lines: 0065
- Output Device: (empty)
- Columns: 0132
- Recipient: (empty)
- Print Immediately

Control Flags:

- Job to Wait for Ext. Program to End

Buttons: OK, Cancel

Through the JSC you can edit the job properties of the real R/3 job.

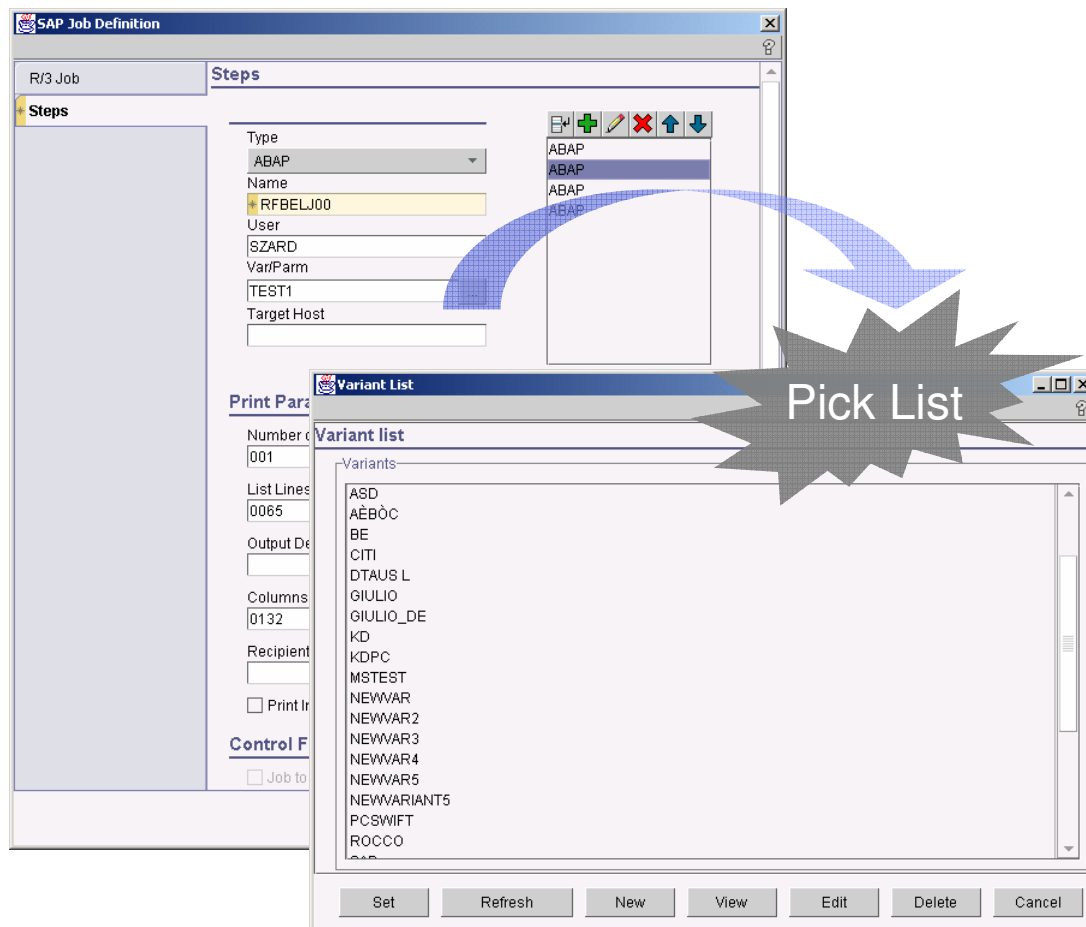
Changes made on this panel actually affect the job properties on the R/3 side, such as the step type, variant or external program parameters.

Two step types are supported:

- ABAP/4
- External Program

When modify a job by adding or removing a step, a new R/3 job is actually created and a new job id is generated.

JSC & XA – R/3 Variant Support



When defining a Standard R/3 job you should specify a variant for each ABAP Step.

The JSC allows you to:

- Query the list of available variants for a specific ABAP.
- Edit/View/Delete an existing variant.
- Create a new variant.

Once a variant is opened, the JSC gets both variant definition and values through r3batch and dynamically builds the proper view for it.

JSC & XA – R/3 Variant Support

Maintain Variants: Report RFEBKA00, Variant NEWVAR

Selection options Variant attributes

File specifications

- Import data
- Elect. bank statement format: A
- Statement file: file1.txt
- Line item file: file2.txt
- PC upload

Posting parameters

- Post immediately Only bank accts
- Generate batch input Session names
- Do not post
- Assign value date

Cash management

- Cash mgmt payment advices Planning type Summarization

Algorithms

- BELNR number interval: 9 to 23
- XBLNR number interval: to
- Bundling: to

Output controls

- Execute as background job
- Print bank statement
- Print posting log
- Print statistics
- List separation

SAP R/3 GUI

Variant Values

File specifications

- Import data
- Elect. bank statement format: A
- Statement file: file1.bt
- Line item file: file2.txt
- PC upload

Posting parameters

- Post immediately Only bank accts
- Generate batch input Session names
- Do not post
- Assign value date

Cash management

- Cash mgmt payment advices Planning type Summarization

Algorithms

- BELNR number interval: I-BT 9 to 23
- XBLNR number interval: I-BQ to
- Bundling: to

Output controls

- Execute as background job
- Print bank statement
- Print posting log
- Print statistics
- List separation

Job Scheduling Console

OK Cancel

JSC & XA – R/3 Variant Support

The image displays two overlapping windows from an SAP R/3 system. The left window is titled "Maintain Variants: Report RFBELJ00, Variant TEST1" and shows a configuration screen for a report variant. It includes fields for Company code (AT01), Document number (10), and Fiscal year (32). Below these are sections for "General selections", "Technical settings", "Further selections", and "Output control", each with various checkboxes and input fields. A yellow starburst graphic with the text "SAP R/3 GUI" is overlaid on this window. The right window is titled "Values" and shows a similar configuration screen for a Job Scheduling Console (JSC) job. It includes fields for Company code (I-WE AT01), Document number (I-ET 10), and Fiscal year (-). It also has sections for "General selections", "Technical settings", "Further selections", and "Output control". A yellow starburst graphic with the text "Job Scheduling Console" is overlaid on this window. Both windows have "OK" and "Cancel" buttons at the bottom right.

JSC & XA – R/3 InfoPackage Jobs

The image shows two SAP dialog boxes. The background window is 'Properties - Job Definition Maestro' with the 'Task' tab selected. It shows 'Task Type' as 'SAP', 'R/3 Job Identity' with 'Job Name' 'ZPAK_3LZ09KEN28O76WK88RLQTCTM' and 'Job ID' 'ipak_', and 'R/3 Job Type' with 'Business Warehouse Info Package' selected. The foreground window is 'SAP Pick List' with 'Name & Location' and 'Look In' sections. The 'Advanced Filter' section has 'InfoSource', 'DataSource', and 'SourceSystem' fields. A table at the bottom shows a list of jobs with columns 'Job Name', 'InfoPackage Name', and 'Job ID'. The first row is selected.

Job Name	InfoPackage Name	Job ID
ZPAK_3LZ09KE...	InfoPackage Test	ipak_
ZPAK_3LZ0E7H...	InfoPackage Tes...	ipak_
ZPAK_3MQ20M9...	MAESTRO_INFO...	ipak_

Total: 3 Displayed: 3 Selected: 1

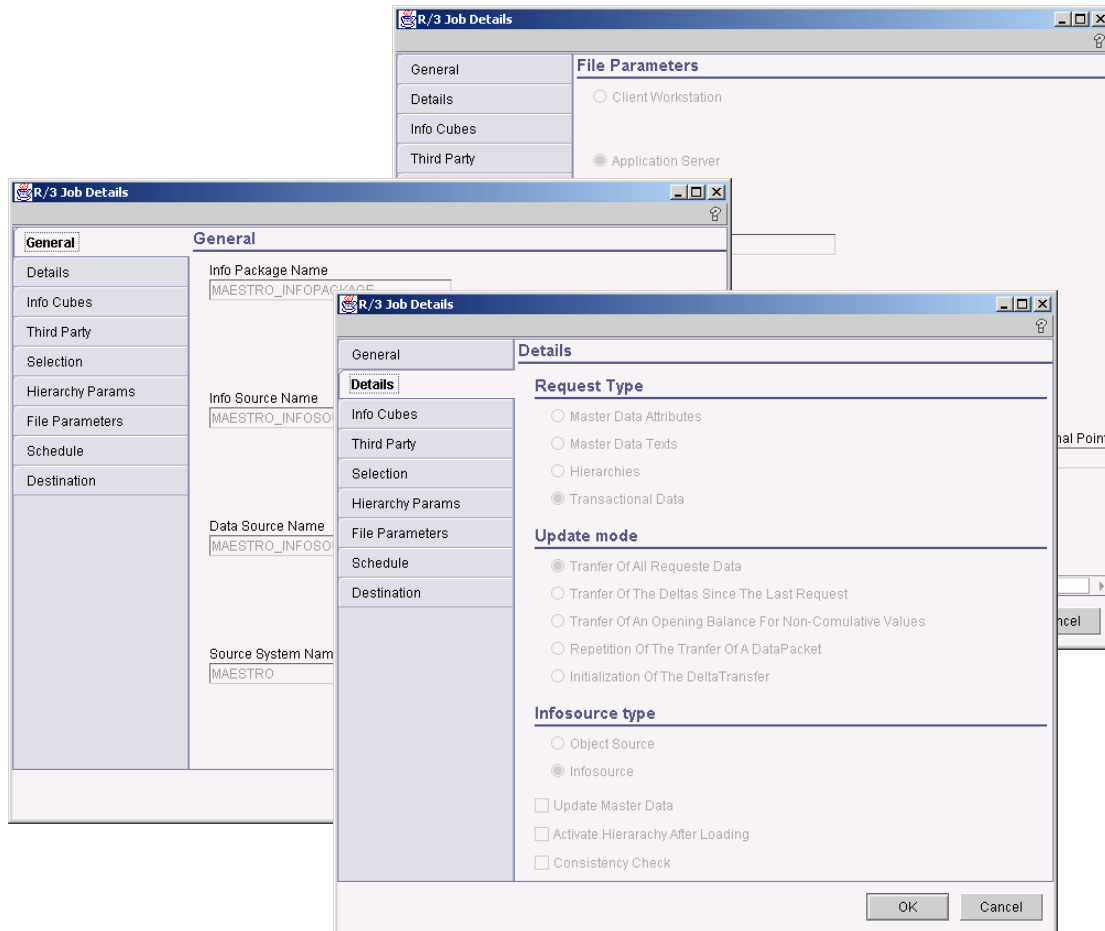
When defining a TWS job whose aim is to run a BW InfoPackage, you should select the target InfoPackage through the pick list task.

In addition to the standard query constraints (name and owner), you can specify advanced filter criteria for the pick list of InfoPackages:

- InfoSource
- DataSource
- SourceSystem

Note that at least SAP Business Information Warehouse 2.0 is supposed to be installed on the target R/3 system.

JSC & XA – R/3 InfoPackage Details

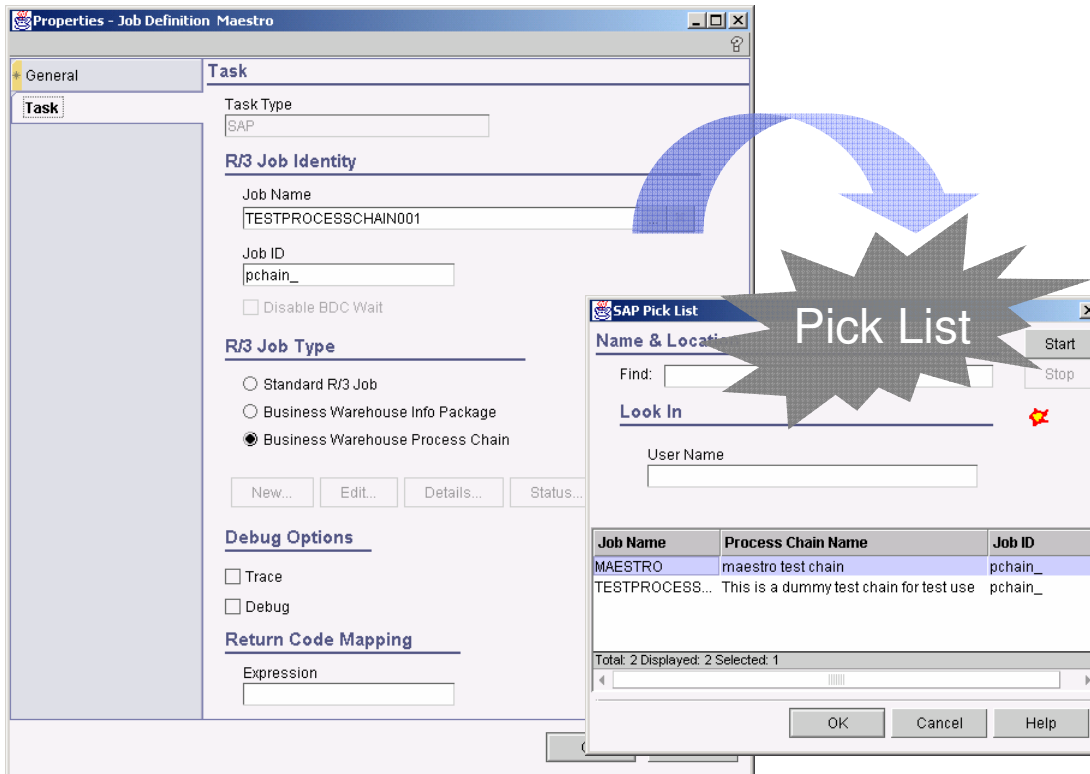


The JSC allows you to get the complete definition of an R/3 InfoPackage.

It contains information such as:

- InfoPackage, InfoSource DataSource names
- Request Type, Update Mode and InfoSource Type
- InfoCube List
- Third Party Selection List
- Selection Options
- Hierarchy Parameters
- File Parameters
- Schedule Options
- Destination Options

JSC & XA – R/3 ProcessChain Jobs



Through the JSC you can get the list of existing Business Warehouse ProcessChains on the target R/3 System.

You can specify either the ProcessChain's name or the owner's one as filter criteria.

Note that at least SAP Business Information Warehouse 3.0 is supposed to be installed on the target R/3 system.

Annonces



IBM Tivoli Workload Scheduler LoadLeveler

- **IBM LoadLeveler has moved to Tivoli !**

- **Product Overview**
- TWS LoadLeveler is a job management system that allows Tivoli customers to optimize workload execution and performance on **AIX and Linux clusters** of systems by matching the jobs processing needs with the available resources.

- TWS LoadLeveler extends the TWS Family to support:
 - ▶ Cluster environments where the customer wants to improve resource utilization and job throughput across a variety of servers.
 - ▶ Cluster environments where the customer wants to exploit idle server cycles.
 - ▶ A Beowulf cluster environment for running massively parallel MPI applications.
 - ▶ Environments where customers are new to cluster computing and need an easy-to-use fully documented and supported job scheduler.
 - ▶ Grid environments in which multiple applications are being dispatched across virtualized resources.

La famille de produits TWS

- **5724-I23 – IBM Tivoli Workload Scheduler LoadLeveler for Linux**
- **5765-E69 – IBM Tivoli Workload Scheduler LoadLeveler for AIX**
- **5698-WSH – IBM Tivoli Workload Scheduler for Distributed**
- **5698-WSE - IBM Tivoli Workload Scheduler for Applications**
- **5724-I64 - IBM Tivoli Workload Scheduler for Virtualized Data Centers**
- **5698-A17 – IBM Tivoli Workload Scheduler for z/OS**

Synthèse

- Architecture « E2E »
- Nouvelle console JSC 8.3
- Interface avec WLM (PTFs)
- Gestion centralisée des sysouts de l'environnement distribué (Tool)
- TWS for Applications (produit)
- Annonces
 - ▶ TWS LoadLeveler





Tivoli software from IBM

Questions ?

