

# PLMJobManagerV2

## Anwender -Dokumentation

Erstellt von: Josef Feuerstein  
Josef Feuerstein

# Inhaltsverzeichnis

<a href="#"><u>Aufgaben des PLMJobManager</u></a>	Folie: 3
<a href="#"><u>Das Funktionsprinzip des PLMJobManager</u></a>	Folie: 4
<a href="#"><u>Die Oberfläche des PLMJobManager</u></a>	Folie: 5
<a href="#"><u>Das Funktionsprinzip des PLMJobManager</u></a>	Folien: 6 - 7
<a href="#"><u>Die Oberfläche des JobServers</u></a>	Folie: 8
<a href="#"><u>Registerkarte Objects (DS/IRS)</u></a>	Folien: 9 - 10
<a href="#"><u>Such Syntax für das selectieren von Daten</u></a>	Folie: 11
<a href="#"><u>Registerkarte Objects (DS/IRS)</u></a>	Folie: 12
<a href="#"><u>JobClients Schedule – Zeiten via JobServer steuern</u></a>	Folien: 13 - 15
<a href="#"><u>Definieren der ProcStateValueRule (Stand 23.11.2011)</u></a>	Folie: 16
<a href="#"><u>Tastaturkürzel (Shortcuts)</u></a>	Folie: 17
<a href="#"><u>Extra Logging – Debuging JobManager Software</u></a>	Folie: 18
<a href="#"><u>Fehlerbeseitigung Font (new 23.12.2011)</u></a>	Folie: 19

# Aufgaben des PLMJobManager

## ■ Ansatz des PLMJobManager

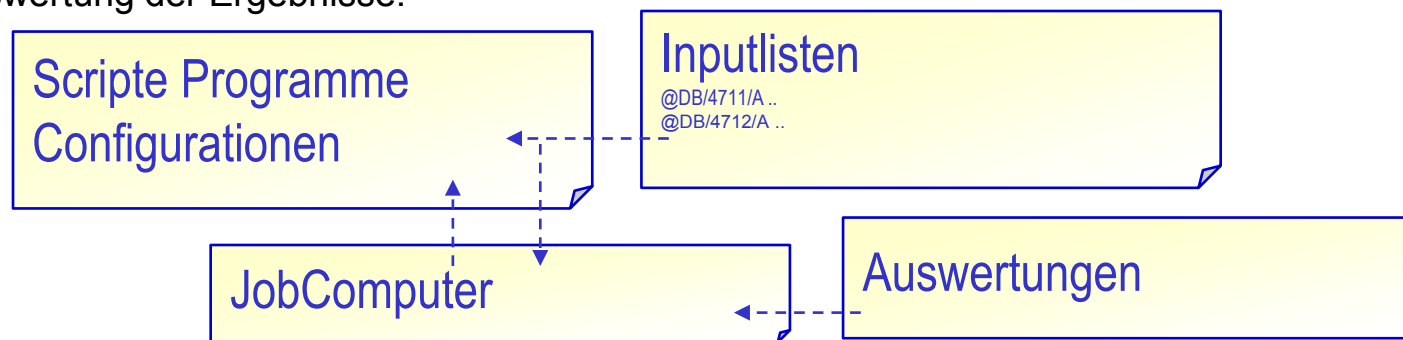
Im NX Umfeld gibt es verschiedenste Programme/Tools um Daten automatisch zu verarbeiten so .z.B.

- NX Refile
- JT-File Erzeugung via → UgToPV
- DXF-File Export via → ugto2d.exe und dxfdwg.exe
- IGES Export
- Step Export
- Parasolid Export
- NX Teilebereinigung ...

Um dieser Programme einsetzen zu können benötigt man:

- Scripte die für die verschiedenen Verarbeitungen mit den Programme deren Konfiguration steuern
- die Listen der zu verarbeitenden Objekte (Abarbeitungs- Reihenfolgen)
- Eine Steuerung der Jobs auf mehreren Rechnern
- eine Auswertung der Ergebnisse.

Schema:



Die Aufgabe des **PLMJobManager** ist es solche Verarbeitungen einheitlich zu Organisieren und zu automatisieren.

# Das Funktionsprinzip des PLMJobManager

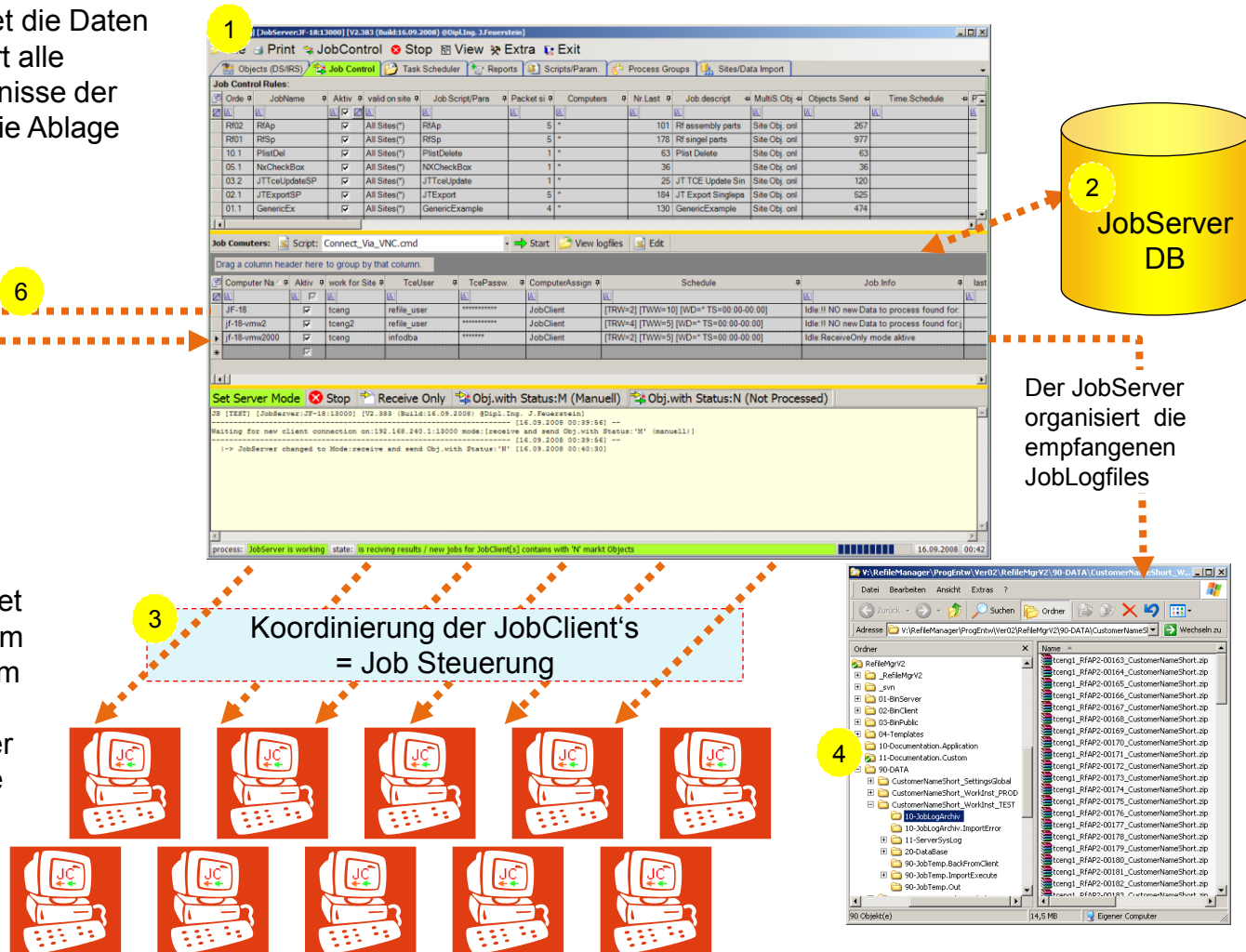
■ JobContol → Übersicht wie arbeitet der 'PLMJobManager'

Der JobServer (1) wird zentral auf einem Arbeitsplatz installiert. Er verwendet die Daten der JobManager-DB (2) und steuert alle JobProzesse (3). Wertet die Ergebnisse der JobProzesse aus und organisiert die Ablage aller Job Logfiles (4).



## Ablauf Jobverarbeitung:

Die JobClients (5) verbinden sich mit dem JobServer (1) und rufen über diesen das nächste JobPacket ab (6). Dieser Job wird nun auf dem JobClient (5) verarbeitet. Nach dem Abschluss der Verarbeitung verbindet sich der JobClient wieder mit dem JobServer und sendet die Ergebnisse an den JobServer (7). Nun ruft der JobClient wieder das nächste JobPacket ab.



# Die Oberfläche des PLMJobManager

## Die Oberfläche unterteilt sich nach den Aufgabenbereichen

- **Objects (DS/IRS)** → Anzeige der Objekt Daten und der Refile Ergebnissen
- **Job Control** → Organisation und Kontrolle der JobClient Prozesse
- **Task Scheduler** → Steuerung von Zeitbezogenen Prozessen
- **Reports** → Auswertung der Job Ergebnisse (Listen Refile-Reports)
- **Scripts/Param.** → Erfassung und Steuerung Job Parameter und Scripte
- **Sites/Data Import** → Erfassung der Sites und Import der Daten (Item's Datasets)

JS [TEST] [JobServer:JF-18:13000] [V2.613 (Build:02.05.2010) (c) Dipl.Ing. J.Feuerstein 2005..2010]

File Print JobControl Stop View Extra Exit

Objects (DS/IRS) Job Control Task Scheduler Reports Scripts/Param. Process Groups Sites/Data Import

Reset Process Result Status: Process Warning Process Results: P.before: 03.05.2010 00:00 P.after: 02.05.2010 00:00

Obj: 720 all OK OK+Err Err Ignor Warnings last res. all res.

Ds/Ir: DS f.Site: tckba (IMC-4901) Own.Site: O.R+S: S OR DsType: Name/Descr.: Release State: Volume: PFM TemplIN:

Ir.Id: BgStrukEx-AGG-Mod-01\_T1/A, BgStrukE O.Name: Nx.Type: all SP AP

PGr: PGr:Rf (Refile) P.Stat: Search & List Ed.P.State View J.Log J.Log J.Link Open NX IRs Mark Extra

Drag a column header here to group by that column.

Ir.Id	O.Name	O.Own	O.Sp	O.UgMA	O.Ug	O.Type	O.PGR:10	P.R	P.TimeStart	P.TimeEnd	P.CMDLine
BgStrukEx-Einzelteil-01_T1/A	BgStrukEx-Einzelteil-01_T1-A	tckba	SP	0	0	UGMASTER	N				
BgStrukEx-Einzelteil-02_T1/A	BgStrukEx-Einzelteil-02_T1-A	tckba	SP	0	0	UGMASTER	N				
BgStrukEx-Einzelteil-03_T1/A	BgStrukEx-Einzelteil-03_T1-A	tckba	SP	0	0	UGMASTER	N				
BgStrukEx-Einzelteil-04_T1/A	BgStrukEx-Einzelteil-04_T1-A	tckba	SP	0	0	UGMASTER	N				
BgStrukEx-Normteil-01_T1/A	BgStrukEx-Normteil-01_T1-A	tckba	SP	0	0	UGMASTER	N				
BgStrukEx-Normteil-02_T1/A	BgStrukEx-Normteil-02_T1-A	tckba	SP	0	0	UGMASTER	N				
BgStrukEx-Normteil-03_T1/A	BgStrukEx-Normteil-03_T1-A	tckba	SP	0	0	UGMASTER	N				
BgStrukEx-KBG-01_T1/A	BgStrukEx-KBG-01_T1-A	tckba	AP	1	3	UGMASTER	N				
BgStrukEx-KBG-03_T1/A	BgStrukEx-KBG-03_T1-A	tckba	AP	5	UGMASTER	N					
BgStrukEx-KBG-02_T1/A	BgStrukEx-KBG-02_T1-A	tckba	AP	2	8	UGMASTER	N				
BgStrukEx-KBG-013_T1/A	BgStrukEx-KBG-013_T1-A	tckba	AP	3	12	UGMASTER	N				

Grand Summaries  
Count = 16

Edit SELECTED rows: [Objects.UltraGrid] Column: [ProcStateColRf]  
 1/1 (100%) Update value of: BgStrukEx-Einzelteil-01\_T1/A/BgStrukEx-Einzelteil-01\_T1-A ProcStateColRf: from: N to: N  
 End Edit SELECTED rows: [Objects.UltraGrid] Column: [ProcStateColRf] duration: [234 [msec]]  
 Info: 0[sec] needed to load and display: 16 datarows visible in DataGrid:16  
 Info: 0[sec] needed to load and display: 16 datarows visible in DataGrid:2  
 Info: 0[sec] needed to load and display: 16 datarows visible in DataGrid:16

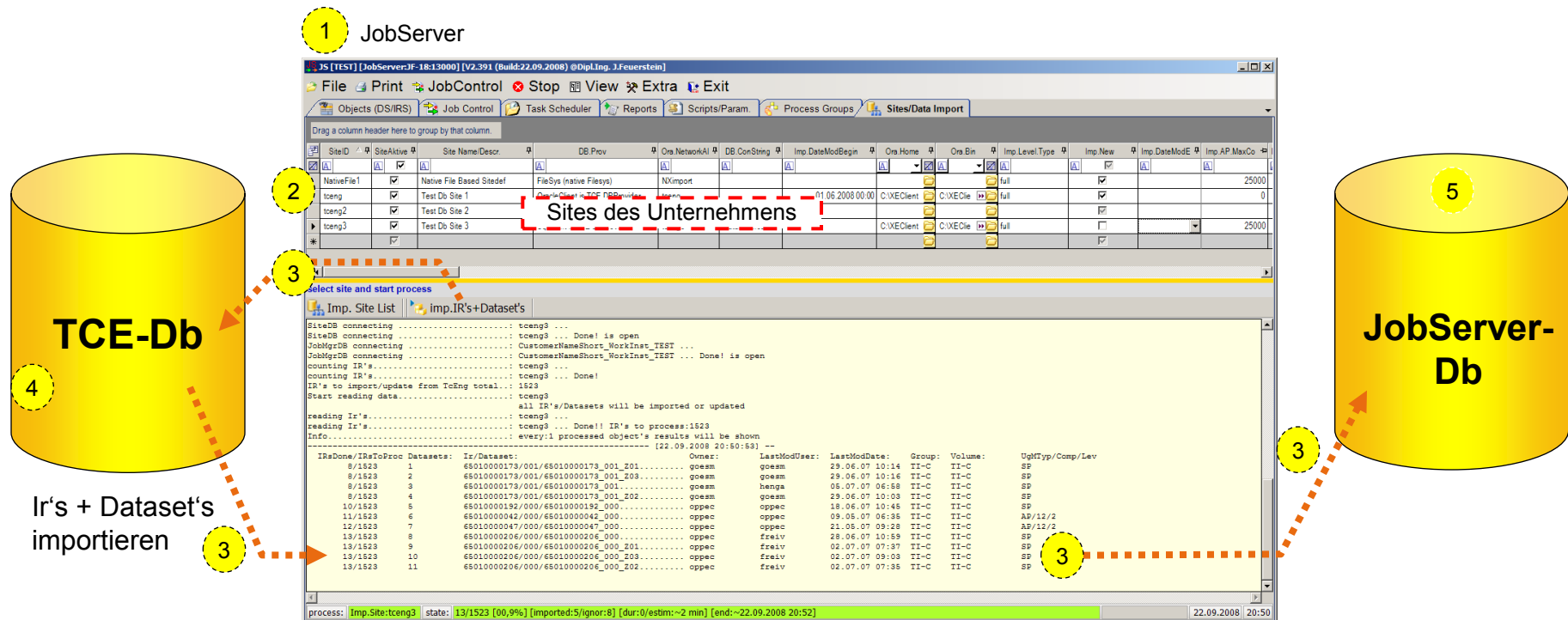
process: idle: state: ... idle .... 02.05.2010 20:05

# Das Funktionsprinzip des PLMJobManager

- TCE Daten aus der TCE Datenbank in die JobServer-DB importieren.

Im JobServer (1) werden die Sites des Unternehmen erfasst (2). Mit der Import/Update (3) Funktion werden aus der TCE-Db (4) die ItemRev und Datasets Daten in die JobServer-DB übertragen.

Diese Daten bilden die JobServer-DB (5) und sind die Datenbasis des JobServer.





# Das Funktionsprinzip des PLMJobManager

- Steuerung der Scripte und Parameter → Konfiguration des JobServers

Im JobServer (1) werden die Scripte (2) und deren Parameter (3) vorgenommen. Zusätzlich werden die erforderlichen Einstellungs-Dateien (4) organisiert.

**1. JobServer Configuration:** The 'Job Scripts and Parameters' table lists various scripts and their parameters.

Job Script Name	JobScriptDescription	Parameters	JobClient Script	JobServer Script to get
Export	Export	-DxIfExpo=YES -ParasolidExpo=YES	NXDataExport\ClientScripts\NXDataExport.cmd	
GenericExmpl	Generic Example	/Para1=Para1-Value /Para2=Para2-Value	GenericExample\ClientScripts\GenericExample.Cmd	
Isam2DToNx	Isam Import		Isam2DToNx\ClientScripts\Isam2D_Convert_to_NX.cmd	InWork\Isam2DToN
JTExport	JT (File Export)	-JTExport=YES	NXDataProcessing\ClientScripts\NXDataProcessing.cmd	
JTtceUpdate	JT (TCE Update)	-JTtceUpdate=YES	NXDataProcessing\ClientScripts\NXDataProcessing.cmd	
NXCheckBox		-keep_volume=yes -update_mod_props=no -non	NXCheckBox\ClientScripts\NXCheckBoxExample.Cmd	
PlistDelete			PlistDelete\ClientScripts\PlistDeleteOnDataSet.cmd	
RIAp	Refile assembly par	-keep_volume=yes -update_mod_props=no -no	RefileNX4\ClientScripts\RefileNX4.Cmd	
RIpSp	Refile single parts	-keep_volume=yes -update_mod_props=no -refile	RefileNX4\ClientScripts\RefileNX4.Cmd	

**2. Script Editor:** The 'RefileNX4.Cmd' script is shown, containing commands for setting environment variables and executing the refile process.

```

set run_refile=%refile_program_exe_DPNE% %JobParameter% -i=%RefilePaket_Dir%
set TceUserPassword=

if exist "%TCEPromptCmd%" Echo TcePrompt Ok: "%TCEPromptCmd%"
if exist "%refile_program_exe_DPNE%" Echo RefileProg Ok: "%refile_program_exe_DPNE%"

if exist "%refile_program_exe_DPNE%" if exist "%TCEPromptCmd%" goto StartRefileError
goto StartRefileError

:StartRefileOk
set title_str=RefileJob: [%JobName%] [%TceSiteId%] [Start: %Date%/%time:~0,6%]
title %title_str%
cho %title_str%
cho -----
cho RefileJob Info:
cho JobName : %JobName%
cho Computer : %ComputerName%
cho RefileDir : %temp%
cho TceSite : %TceSiteId%
cho TceUser : %TceUser%
cho.
cho RefileOpt : %JobParameter:~0,65%
cho Sta
cho.

```

**3. Setup refile parameters:** The 'Setup refile parameters' dialog box is shown, allowing users to configure various options for the refile process.

Type	Option	Value	Description
<input checked="" type="checkbox"/>	-keep_volume=	yes	<yes/no> use original volume for refilled parts
<input checked="" type="checkbox"/>	-update_mod_props=	no	<yes/no> update last modifying user/date on dataset
<input type="checkbox"/>	-drawing=	no	<yes/no> alle views in drawings will be updated
<input checked="" type="checkbox"/>	-non_masters=	yes	<yes/no> automatically refile non-master parts
<input checked="" type="checkbox"/>	-refile_released=	yes	<yes/no> refile parts with release status
<input checked="" type="checkbox"/>	-bypass=	yes	<yes/no> use bypass privilege if necessary
<input checked="" type="checkbox"/>	-structure_sync=	no	<no/for_imanfrom_ug> synchronize structures during refile
<input checked="" type="checkbox"/>	-force_structure_sync=	no	<yes/no> force synchronize structures during refile as requested by -structure_sync=
<input type="checkbox"/>	-y		Fully load assembly, and refills components that are not at the current version of NX
<input type="checkbox"/>	-force_refile		Used with the -y switch to force all components to be refilled
<input type="checkbox"/>	-transforms_source=		<from_imanfrom_ug> source of transforms if different from -structure_sync argument
<input type="checkbox"/>	-generate_trueshape=	no	<yes/no> generate True Shape occupancy data without refilling parts
<input type="checkbox"/>	-record_part_data=	no	<yes/no> record part data (Bounding Box, Mass Properties) without refilling parts
<input type="checkbox"/>	-record_altprep_notes=	no	<yes/no> record AltRep occurrence notes (temporary option) without refilling parts

**4. File Explorer:** The file explorer shows the directory structure for the refile process, including the '90-DATA' folder and the 'CustomerNameShort\_SettingsGlobal' folder.

# Die Oberfläche des JobServers

## ■ Details der Oberfläche des JobServers

Die Oberfläche des JobServers ist nach den Hauptaufgaben in Registerkarten (1) aufgeteilt. Jede Registerkarte teilt sich auf nach: „Funktionsbereich (2)“ dem „Listenbereich (3)“ „dem Textausgabefenster (4)“ sowie einer „Statusleiste (5)“. Die Software wurde in Ihren Grundlagen so konzipiert, dass in allen Bereichen eine möglichst gleichartige Bedienung gewährleistet ist.

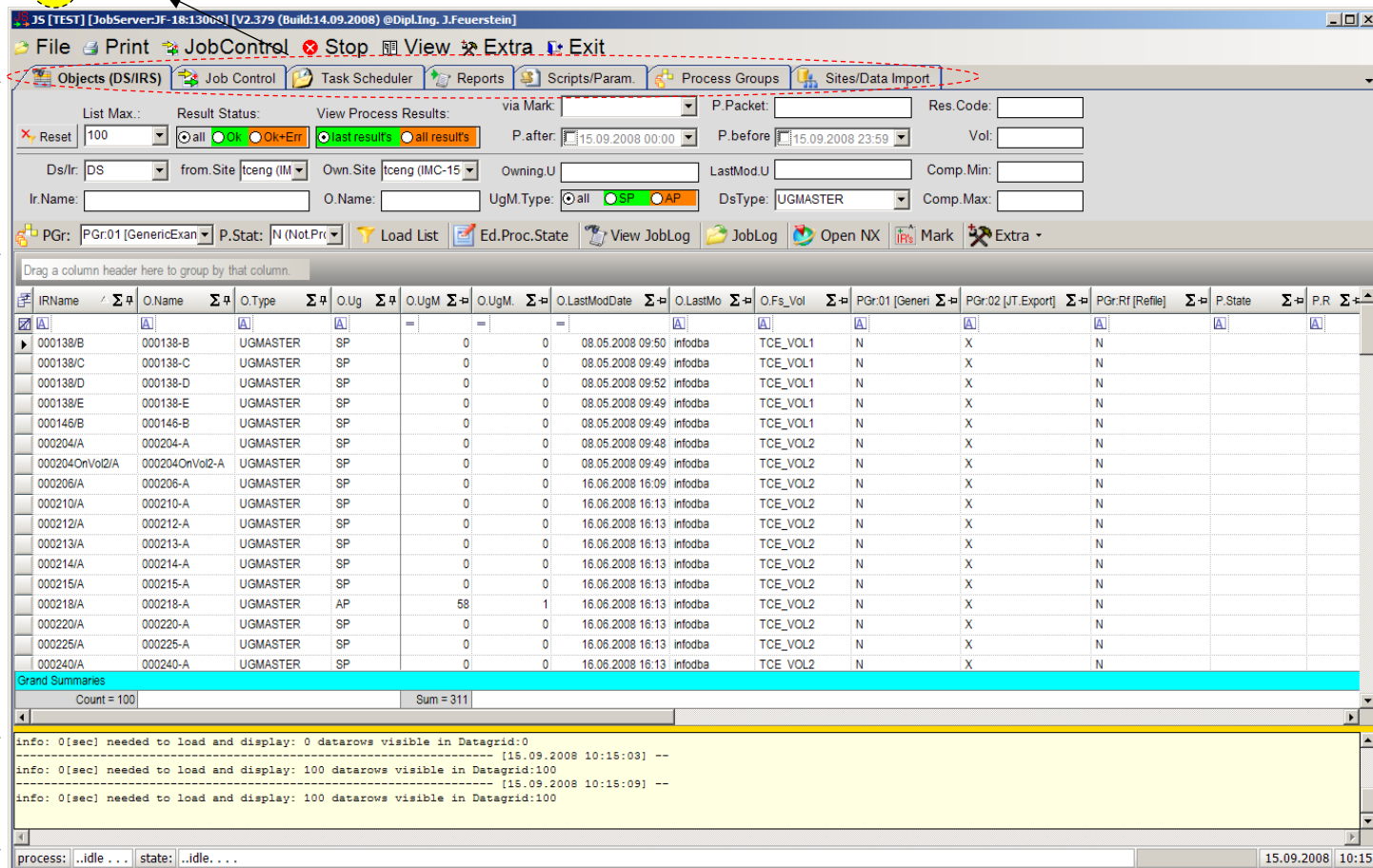
### 1 Registerkarten

### 2 Funktionsbereich

### 3 Listenbereich

### 4 Textausgabefenster

### 5 Statusleiste



JS [TEST] [JobServer:JF-18:1306S] [V2.379 (Build:14.09.2008) @Dipl.Ing. J.Feuerstein]

File Print JobControl Stop View Extra Exit

Objects (DS/IRS) Job Control Task Scheduler Reports Scripts/Param. Process Groups Sites/Data Import

List Max.: 100 Result Status: all OK OK+Err last results: all results View Process Results: via Mark: P. Packet: Res. Code: Vol: P. after: 15.09.2008 00:00 P. before: 15.09.2008 23:59

Ds/Ir: DS from Site: tceng (IM) Own Site: tceng (IMC-15) Owing U: LastMod U: Comp. Min: Ir Name: O Name: UGM Type: all SP AP Ds Type: UGMASTER Comp. Max:

PGr: PGr:01 [GenericExan] P.Stat: N (NotPr) Load List Ed.Proc.State View JobLog JobLog Open NX IRs Mark Extra

Drag a column header here to group by that column.

IRName	OName	OType	O Ug	O UgM	O UgM	O LastModDate	O LastMo	O Fs_Vol	PGr:01 [Generi	PGr:02 [T.Export	PGr:Rf [Refle]	P.State	P.R
000138/B	000138-B	UGMASTER	SP	0	0	08.05.2008 09:50	infodba	TCE_VOL1	N	X	N		
000138/C	000138-C	UGMASTER	SP	0	0	08.05.2008 09:49	infodba	TCE_VOL1	N	X	N		
000138/D	000138-D	UGMASTER	SP	0	0	08.05.2008 09:52	infodba	TCE_VOL1	N	X	N		
000138/E	000138-E	UGMASTER	SP	0	0	08.05.2008 09:49	infodba	TCE_VOL1	N	X	N		
000146/B	000146-B	UGMASTER	SP	0	0	08.05.2008 09:49	infodba	TCE_VOL1	N	X	N		
000204/A	000204-A	UGMASTER	SP	0	0	08.05.2008 09:48	infodba	TCE_VOL2	N	X	N		
000204OnVol2/A	000204OnVol2-A	UGMASTER	SP	0	0	08.05.2008 09:49	infodba	TCE_VOL2	N	X	N		
000206/A	000206-A	UGMASTER	SP	0	0	16.06.2008 16:09	infodba	TCE_VOL2	N	X	N		
000210/A	000210-A	UGMASTER	SP	0	0	16.06.2008 16:13	infodba	TCE_VOL2	N	X	N		
000212/A	000212-A	UGMASTER	SP	0	0	16.06.2008 16:13	infodba	TCE_VOL2	N	X	N		
000213/A	000213-A	UGMASTER	SP	0	0	16.06.2008 16:13	infodba	TCE_VOL2	N	X	N		
000214/A	000214-A	UGMASTER	SP	0	0	16.06.2008 16:13	infodba	TCE_VOL2	N	X	N		
000215/A	000215-A	UGMASTER	SP	0	0	16.06.2008 16:13	infodba	TCE_VOL2	N	X	N		
000218/A	000218-A	UGMASTER	AP	58	1	16.06.2008 16:13	infodba	TCE_VOL2	N	X	N		
000220/A	000220-A	UGMASTER	SP	0	0	16.06.2008 16:13	infodba	TCE_VOL2	N	X	N		
000225/A	000225-A	UGMASTER	SP	0	0	16.06.2008 16:13	infodba	TCE_VOL2	N	X	N		
000240/A	000240-A	UGMASTER	SP	0	0	16.06.2008 16:13	infodba	TCE_VOL2	N	X	N		

Grand Summaries  
Count = 100 Sum = 311

info: 0[sec] needed to load and display: 0 datarows visible in Datagrid:0  
----- [15.09.2008 10:15:03] ---  
info: 0[sec] needed to load and display: 100 datarows visible in Datagrid:100  
----- [15.09.2008 10:15:09] ---  
info: 0[sec] needed to load and display: 100 datarows visible in Datagrid:100

process: ..idle... state: ..idle... 15.09.2008 10:15



# Registerkarte Objects (DS/IRS)

## ■ Funktions-Beschreibung der Registerkarte 'Objects(DS/IRS)'

Über diese Oberfläche können die in der Datenbank befindlichen IR/ Dataset Objekte zur Anzeige gebracht werden. Im Funktionsbereich (1) dieser Registerkarte können Filterbedingungen für die anzuzeigenden Daten angegeben werden. Die Filterbedingungen unterteilen sich nach Filterbedingungen für Prozessdaten (2) (z.B. Processed after) und nach Filterbedingungen für Objektdaten (3) und filtern nach. Die Filterbedingungen arbeiten nach dem Prinzip der Daten einschließenden Filterungen. D.h. es werden nur Daten zur Anzeige gebracht die den Filterbedingungen entsprechen.

**Allgemeines Funktionsprinzip:** Filter angeben und den Schalter „Load List“ (4) anwenden anschließend werden die Daten im Listebereich (5) angezeigt.

1 Funktionsbereich

2 Eingabebereich für Prozessdatenfilter

3 Eingabebereich für Objektdatenfilter

4 Load List

5 Listebereich

IRName	O.Name	O.Type	O.Ug	O.UgM	O.LastModDate	O.LastModUser	O.Fs_Vol	PGr.01	PGr.02
000002/A	000002-A	UGMASTER	SP	0	27.07.2008 08:38	infodba	TCE_VOL1	N	X
	Z1	UGPART	SP	0	27.07.2008 08:38	infodba	TCE_VOL1	N	X
000008/A	000008-A	UGMASTER	SP	0	08.05.2008 10:24	infodba	TCE_VOL1	N	X
000009/A	000009-A	UGMASTER	AP	2	21.01.2008 11:11	refile_user	TCE_VOL1	N	X
BgStrukEx01/A	BgStrukEx01-A	UGMASTER	AP	3	06.05.2008 11:28	infodba	TCE_VOL2	N	N
BgStrukEx01_02/A	BgStrukEx01_02-A	UGMASTER	AP	3	04.05.2008 21:46	infodba	TCE_VOL2	N	N
BgStrukEx-AGG-Mod-01_01/A	BgStrukEx-AG...	UGMASTER	AP	36	21.01.2008 11:21	refile_user	TCE_VOL1	N	N
BgStrukEx-Einzelteil-01_01/A	BgStrukEx-Einz...	UGMASTER	SP	0	08.05.2008 09:49	infodba	TCE_VOL1	N	N
BgStrukEx-Einzelteil-02_01/A	BgStrukEx-Einz...	UGMASTER	SP	0	08.05.2008 09:49	infodba	TCE_VOL2	N	N
BgStrukEx-Einzelteil-03_01/A	BgStrukEx-Einz...	UGMASTER	SP	0	08.05.2008 09:49	infodba	TCE_VOL1	N	N

Grand Summaries  
Count = 27 Sum = 164

JS [TEST] [JobServer:JF-18:13000] [V2.391 (Build:22.09.2008) @Dipl.Ing. J.Feuerstein]  
[23.09.2008 04:46:39] --  
[23.09.2008 04:46:39] --  
info: 3[sec] needed to load and display: 27 datarows visible in Datagrid:27

process: ..idle ... state: ..idle ... 23.09.2008 06:35

# Registerkarte Objects (DS/IRS)

## ■ Übersicht über die 'Filterfelder'

Setzt alle Filter auf Standard Werte

Begrenzt auf max. zu ladende Obj.

Filter nach Result Status  
 all = Status wird ignoriert (Standard)  
 Ok = nur Obj. mit ResultCode=0  
 Ok\*Err = nur Obj. die einen ResCode haben  
 Err = nur Obj. mit ResultCode<>0

Schaltet um zwischen filtern der Objektprozess Daten  
 - nur letztes Ergebnis  
 - aller Ergebnisse des

Setzt das filtern auf Objekte die **nach** dem angegebenen **Datum** **verarbeitet** wurden

Setzt das filtern auf Objekte die **vor** dem angegebenen **Datum** **verarbeitet** wurden

Setzt das filtern nach Res. Code z.B. 970002

Setzt das filtern nach Obj. Tce Volume z.B. TcVol01

Hinweis: das Control Kästchen setzt bzw. deaktiviert diesen Filter

Objects (DS/IRS) Job Control Task Scheduler Reports Scripts/Param. Process Groups Sites/Data Import

List Max.: 100 Result Status: ☒ all ☒ Ok ☐ Ok+Err ☐ Err View Process Results: ☒ last result's ☐ all result's

via Mark: P.Packet: Res.Code: P.after: 23.09.2008 00:00 P.before: 23.09.2008 23:59 Vol: Ds/Ir: DS from Site: tceng (IMC-1500) Own Site: \* Owing.U: LastMod.U: Comp.Min: Ir.Name: 000002/A;BgStrukEx-KBG-03\_01/A; O.Name: UgM.Type: ☒ all ☒ SP ☐ AP DsType: Comp.Max: PGr: PGr:02 [JT.Export] P.Stat: Load List Ed.Proc.State View JobLog JobLog Open NX IR's Mark Extra

Drag a column header here to group by that column.

IRName	O.Name	O.Type	O.Ug	O.UgM	O.LastModDa	O.LastModU	PGr:01 [Generic]	PGr:02 [JT.Exp]	P.ResultCo	P.ResultMsg
000002/A	000002-A	UGMASTER	SP	0	27.07.2008 08:38	infodba	N	X		
	Z1	UGPART	SP	0	27.07.2008 08:38	infodba	N	X		

Setzt Obj.Filter auf Ir = Filtern auf ItemRev.  
 DS = Dataset

Setzt Obj.Filter auf Ir Name  
 Bspl: 4711 → Anzeige aller DS des Ir's 4711\*  
 Bspl: 4711/A; Anzeige alle DS von Ir 4711/A  
 Bspl: 4711/A;00234/A; alle DS von Ir 4711/A und 00234/A  
 Bspl: 4711/\*;00234/\*; alle DS von Ir 4711/\* und 00234/\*

Setzt Obj.Filter auf den Namen des Objektes  
 Bspl: Z1 → Anzeige aller DS die im Namen Z1\* haben

Setzt Obj.Filter nach  
 SP = Single Parts = Einzelteil  
 AP = Assembly Part = Baugruppe  
 All = Sp + AP (Standard)

Hinweis: Das Semikolon ; wird zum trennen der IR's verwendet



Tip:

Sollten nicht die erwarteten Daten angezeigt werden, so drücken Sie den Schalter „Reset“ und geben Sie die Filterbedingungen erneut ein und drücken Sie dann wieder den Schalter „Load List“. Hintergrund: In der Praxis kommt es häufiger vor, dass noch eine Filterbedingung mehr gesetzt ist als erwartet. Dieser wird durch den Reset Schalter auf Standard gesetzt, so dass dann mit den geänderten Filterbedingungen die zu erwartenden Daten geladen werden.

# Such Syntax für das selectieren von Daten

- Dokumentation der '**Such-Syntax**' für das Auflisten oder selektieren von IR/DS Daten

## 'Such-Syntax'

Im JobManager gelten allgemeine Suchregeln für das Listen von Daten.

Diesen gelten für die Eingabefelder beim auflisten von DS/IRS. Wie auch beim Definieren der Datenselection für eine JobControl definition.

Hierfür wurde die folgende allgemeine Suchsyntax erstellt.

### Beispiel: Listen von Object Daten

Eingabe: 4711/\*;00234/\*;

|→ listet alle DS von Ir 4711/\* und 00234/\*

### Beispiel Volumes:

Eingabe: vol1;vol2; → sucht nach allen Daten die dem vol1 und vol2 gehören

Eingabe: vol1;vol2

|→ sucht nach allen Daten die genau den wert **vol1** oder **vol2** haben.

|→ Daten die andere Werte sind ausgeschlossen

### Erweiterte Regeln mit Excludes

Eingabe: Vol\***!not!**vol2;vol3;volLocation2\*

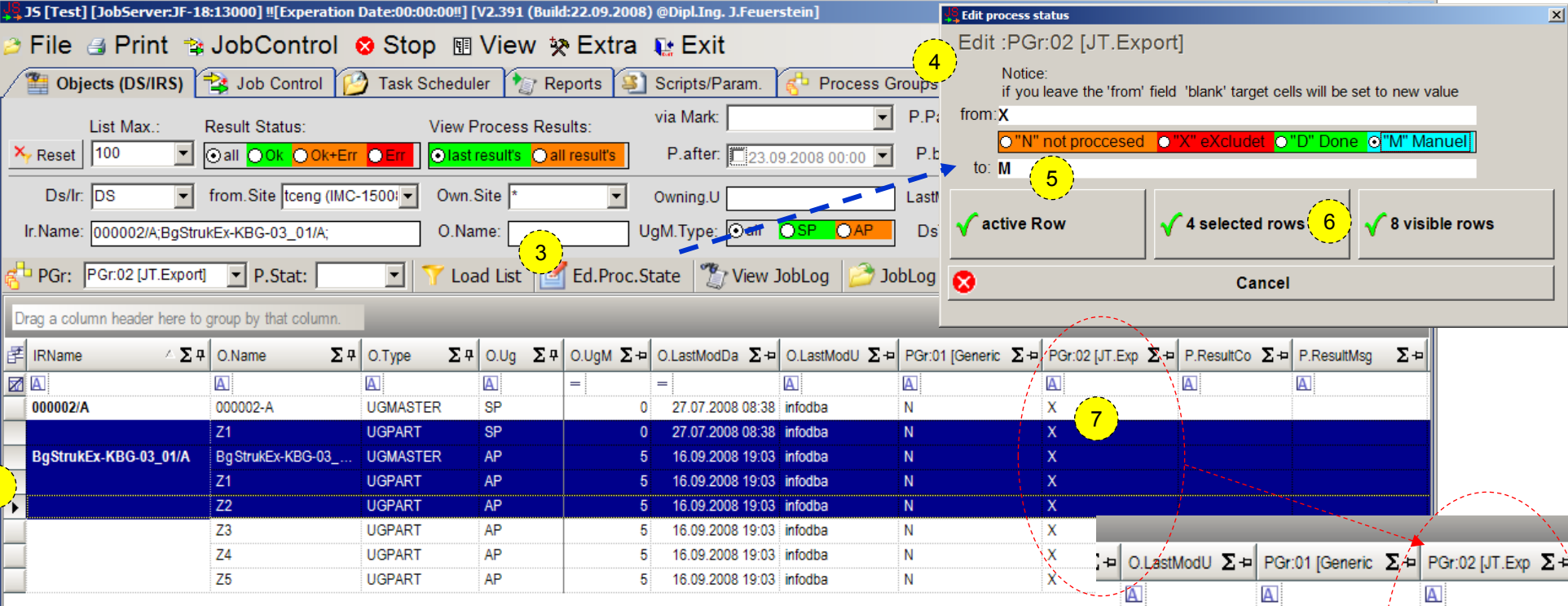
|→ sucht nach allen Daten die mit **vol\*** begingen.

|→ Daten die den Wert **vol2** oder **vol3** oder **volLocation2\*** haben werden ausgeschlossen

# Registerkarte Objects (DS/IRS)

## Funktionsleiste Editieren des 'Prozesstatus'

Durch das Editieren des Protzesstatus können Objekte zur Prozessverarbeitung hinzugefügt werden.



**1** PGr: [PGr:02 [JT.Export]]

**2** Ir.Name: 000002/A;BgStrukEx-KBG-03\_01/A;

**3** O.Name: [ ]

**4** Edit :PGr:02 [JT.Export]

Notice:  
if you leave the 'from' field 'blank' target cells will be set to new value

from: X

to: M **5**

**6** 4 selected rows

**7** X

**8** M

**Vorgehensweise:**

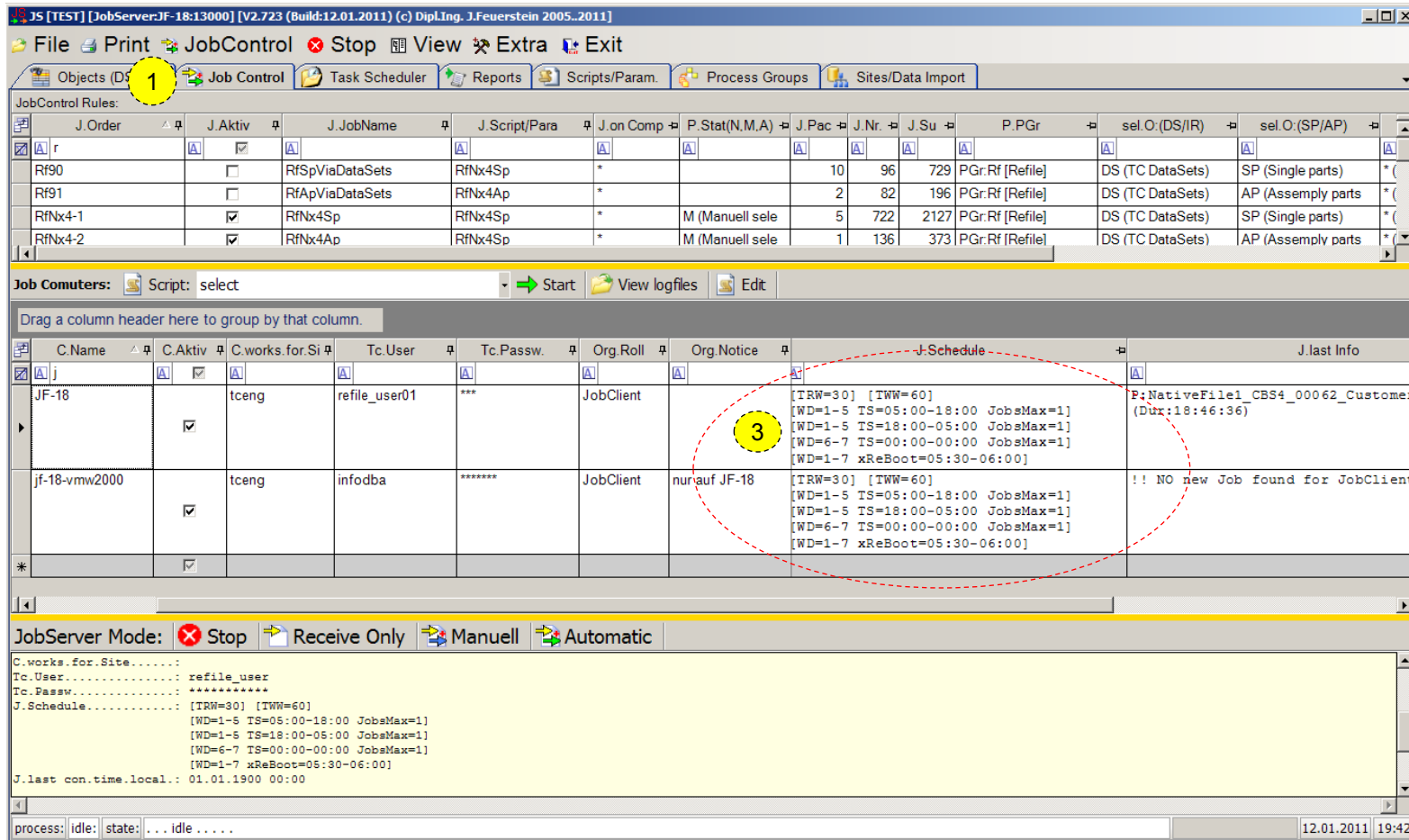
- Achten sie darauf das die Richtige Prozessgruppe (**1**) ausgewählt wurde
- Selektieren die die Objekte deren Status umgesetzt werden soll (**2**)
- Klicken Sie den Schalter „Ed.Proc.Stat“ (**3**)
- Geben sie im folgenden Dialog (**4**) den gewünschten neuen Zielstatus ein Bspl. „M“ (**5**)
- Klicken Sie den Mittleren Schalter „n selected rows“ (**6**)
- Ergebnis die Editierten Objekt Prozess Status (**7**) → (**8**)

IRName	O.Name	O.Type	O.Ug	O.UgM	O.LastModDa	O.LastModU	PGr:01 [Generic]	PGr:02 [JT.Exp]	P.ResultCo	P.ResultMsg
000002/A	000002-A	UGMASTER	SP	0	27.07.2008 08:38	infodba	N	X		
BgStrukEx-KBG-03_01/A	BgStrukEx-KBG-03_...	UGPART	AP	5	16.09.2008 19:03	infodba	N	X		
	Z1	UGPART	AP	5	16.09.2008 19:03	infodba	N	X		
	Z2	UGPART	AP	5	16.09.2008 19:03	infodba	N	X		
	Z3	UGPART	AP	5	16.09.2008 19:03	infodba	N	X		
	Z4	UGPART	AP	5	16.09.2008 19:03	infodba	N	X		
	Z5	UGPART	AP	5	16.09.2008 19:03	infodba	N	X		

# JobClients Schedule – Zeiten via JobServer steuern

## ■ Steuerung der JobClient Schedule Zeiten

Der JobServer steuert die Zeiten in dem ein JobClient aktive ist über JobControl (1) → Job Computers (2) J.Schedule (3)



The screenshot displays the JobServer interface with the following components:

- JobControl Rules (1):** A table listing job rules with columns for J. Order, J. Aktiv, J. JobName, J. Script/Para, J. on Comp, P. Stat(N,M,A), J. Pac, J. Nr, J. Su, P. PGr, sel. O: (DS/IR), and sel. O: (SP/AP).
- Job Computers (2):** A table listing job computers with columns for C. Name, C. Aktiv, C. works for Si, Tc. User, Tc. Passw, Org. Roll, Org. Notice, J. Schedule, and J. last Info.
- J.Schedule (3):** A detailed view of the job schedule for a specific job client, showing time slots and job execution parameters.

The JobServer Mode is set to **Stop**. The JobServer Mode options are: Stop, Receive Only, Manuell, and Automatic.

The JobServer Mode is set to **Stop**. The JobServer Mode options are: Stop, Receive Only, Manuell, and Automatic.

The JobServer Mode is set to **Stop**. The JobServer Mode options are: Stop, Receive Only, Manuell, and Automatic.

# JobClients Schedule – Zeiten via JobServer steuern

## ■ Steuerung des der JobClient Schedule Zeiten

Steuerung der JobClient via ' **JobClientScheduleControl** '

### Wochentag Steuerung (Key:WD WeekDay):

[WD=\* ] alle Tage der Woche

[WD=1-7] alle Tage der Woche

[WD=1-3] alle Tage der Woche Montag – Mittwoch

[WD=1-3-Montag-Mittwoch] die Tage von Montag bis Mittwoch also inc Dienstag

[WD=6-7-Sa-So] die Tage von Samstag und Sonntag

[WD=3-Mi-Fron] nur der Mittwoch

### Zeitpunkt (Key:TP TimePoint):

[WD=1-7 TP=21:13] wird dann als OK zurückgegeben wenn die aktuelle Zeit in Std und Min übereinstimmt

TWR= Dauer in Sec die der JobClient warten soll wenn er innerhalb der Zeit liegt bevor er sich wieder mit dem Server verbindet

Beispiel: [TWR=30]TWW= Dauer in

Sec die der JobClient warten soll wenn er außerhalb der Zeit liegt bevor er sich wieder mit dem Server verbindet

Beispiel: [TWW=60]

### Zeitspanne: (Key:TS TimeSpan =TS.Begin-TS.End))

[WD=\* TS=00:00-00:00] [WD=1-Mo TS=20:30-05:00]

[WD=1-3 TS=10:30-03:00] [WD=4-7 TS=08:30-03:00]

[WD=1-5-Mo-Fr TS=19:15-05:30] [WD=3-Mi-Fron TS=19:15-23:59] [WD=4-Do-Fron TS=00:00-17:30] [WD=6-7-Sa-So TS=00:00-00:00]

### Anzahl Jobs (Key:JobsMax short Key:JMax):

definiert wie viele Jobs auf dem Client im angegebenen Zeitbereich gleichzeitig abgearbeitet werden dürfen.

Hinweis: Wird mit JobsMax=0 angegeben so wird KEIN Job im angegebene Zeitbereich verarbeitet **Beispiel.:** [WD=1-5 TS=07:00-17:30 **JobsMax=0**]

Beispiel: [WD=6-7 TS=00:00-00:00 **JobsMax=2**] definiert das am Samstag und Sonntag zu allen Zeiten maximal 2 Jobs auf dem Client laufen

### Job Resoure Steuerung (Key:JobsResPointsMax short Key: JrpMax):

definiert wie viele Resoure Punkte ein Client hat. Diese Zahl wird verglichen mit der Anzahl der Resourepunkte die sich aus der Summe der Resourepunkte der auf dem Clienten laufenden Jobs ergeben (**ResourcePointsOfCurExecutingJobs**). Wenn  $\text{ResourcePointsOfCurExecutingJobs} + \text{JobResourcePoints} > \text{JobsResPointsMax}$  so erhält dieser Client keine Jobs

Beispiel: [WD=6-7 TS=00:00-00:00 JobsMax=2 JobsResPointsMax=2]

### Der JobClient Computer automatisch neu starten (Key:Reboot:)

Beispiel: [WD=1-7 ReBoot=05:30-06:00] definiert das sich der Rechner an den allen Wochentage in der Zeit zwischen 05:30 und 06:00 bootet wenn in dieser Zeitspanne keine Job mehr bearbeitet wird.



# JobClients Schedule – Zeiten via JobServer steuern

## ■ Steuerung der JobClient Schedule Zeiten

### Example: Standard settings:

```
[TRW=30] [TWW=60]  
[WD=1-5 TS=05:00-18:00 JobsMax=1 JobsResPointsMax=-1]  
[WD=1-5 TS=18:00-05:00 JobsMax=1 JobsResPointsMax=-1]  
[WD=6-7 TS=00:00-00:00 JobsMax=1 JobsResPointsMax=-1]  
[WD=1-7 xReBoot=05:30-06:00] Hinweis wird xReBoot=... angegeben so wird ReBoot ignoriert
```

# Definieren der ProcStateValueRule (Stand 23.11.2011)

## ■ Steuerung des Prozess Ergebnisses ('ProcStateValueRule')

### Definieren der : 'ProcStateValueRule'

Über die ProcStateValueRule können Regeln festgelegt werden wie der P.Status Wert in Abhängigkeit vom Job Ergebnis sich festlegt. Ausgewertet wird dabei die Daten des JobErgebnisses: ResultCode ResultMessage ResultWarning

#### ResultMessage: (Key:O.Type) Optional

werte den Object Typen (UGMASTER,UGPART,UGALTREP..) aus es ermöglicht das

#### ResultCode: (Key:J.ResCode) Erforderlich

es kann definiert werden: RC=0 RC>0 RC<>0 RC<0

#### ResultMessage: (Key:J.ResMsg) Erforderlich

Die der Wert der JobMessage wird via Win Like Methode verglichen Beispiel: RM=\* oder RM=\*Corba\*

#### ResultWarning: (Key:J.ResWarn) Erforderlich

Der Wert von JobWarning (Boolean) 0=false 1=true verglichen ResWarn=\* (true da Ignoriert) RW=1 (true has Waring) oder RW=0 (false no Warning)

#### Result Process Status: (Key:P.State)

Definiert den Wert den das Objekt erhält wenn eine der Regeln zutrifft.

Die Auswertung der ProcStateValueRule wird beendet sowie eine der Regeln zutrifft. Der das Ergebnis P.Stat=... wird dann in dann als P.Status übergeben. Des weiteren gilt das wenn keine der Regeln zutreffen P.Status = D (Done) gesetzt wird

### Beispiel Syntax der Definition:

Jede Definition steht in [] Klammern und kann aus mehreren Regeln bestehen

Default Values: „..\01-BinServer\Resources\ApplicationStandards\ProcStatValueRule\_Template.TXT“

```
[RuleID=Default] [P.Stat=D]
[RuleID=OK]      [O.Type=UGMASTER] [J.ResCode=0] [J.ResMsg=*) [J.ResWarn=0] [P.Stat=D.RfBa.OK]
[RuleID=OK]      [O.Type<>UGMASTER] [J.ResCode=0] [J.ResMsg=*) [J.ResWarn=0] [P.Stat=D.OK]
[RuleID=Err]     [O.Type=*) [J.ResCode>0] [J.ResMsg=*) [J.ResWarn=*) [P.Stat=D.RfBa.ERR]
[RuleID=Warn]    [O.Type=*) [J.ResCode=*) [J.ResMsg=*) [J.ResWarn=1] [P.Stat=D.RfBa.W]
```

# Tastaturkürzel (Shortcuts)

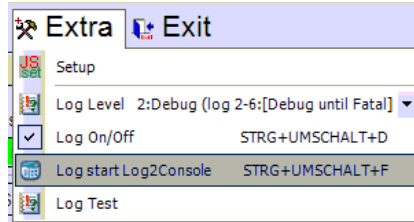
## ■ Auflistung der Tastaturkürzel

Nr .	Situation / Kürzel	Beschreibung:	Beispiel
1	JobServer startet <b>Hold Shift Key</b>	Software wird im Setup Modus gestartet. Nach dem Start der Software öffnet sich der Settingsdialog.	
2	JobServer startet <b>Hold Keys Shift + Strg</b>	Das Splashscreen Fenster gibt erweiterte Infos aus und bleibt nach dem Start der Application geöffnet. Evtl. gesetzte AutoStart Optionen werden nicht ausgeführt.	
3	JobServer Tabellenansicht <b>Strg+I</b>	Kopiert die Daten der aktiven Datenzeile in das Infofenster	
4	JobServer <b>Strg+S</b>	Speichert die Aktuellen Einstellungen wie Fensterpositionen Tabellenansichten und aktuelle Eingabedaten der Oberfläche	
5	JobServer Tabellenansicht <b>Strg+R</b>	Passt die Höhe der Tabellenzeilen an die Texthöhen der in den Tabellenzeilen befindlichen Daten an. Hierdurch können Datenzeilen Inhalte sehr schell sichtbar gemacht werden	

# Extra Logging – Debugging JobManager Software

■ Working with Debug window „Log2Console“ This Tool can be used for debugging the PLMJobManager software programm

Start the Software via Extra → Log start Log2Console (STRG+UMSCHALT+F) (1)

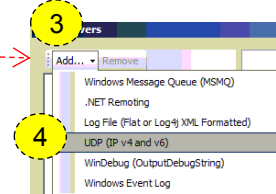
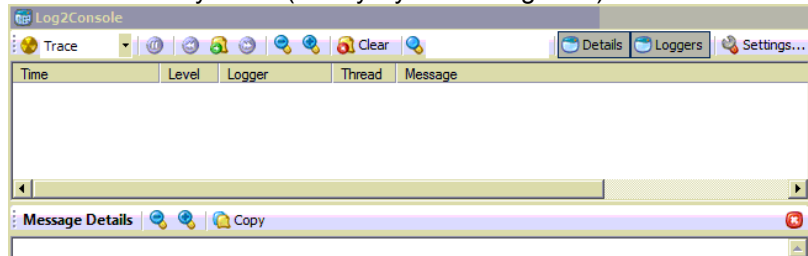


First time you use the Software on your Computer, you have to set settings in Log2Console Application:

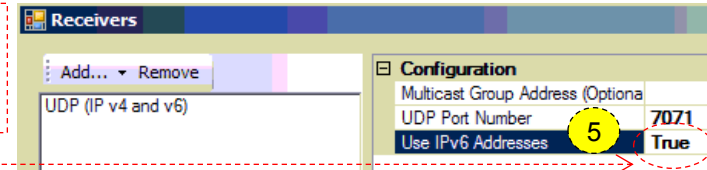
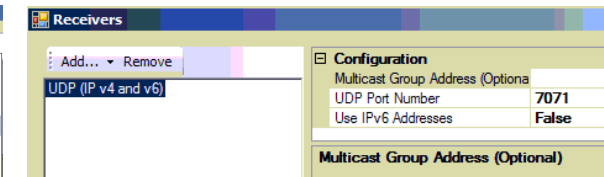


## Description:

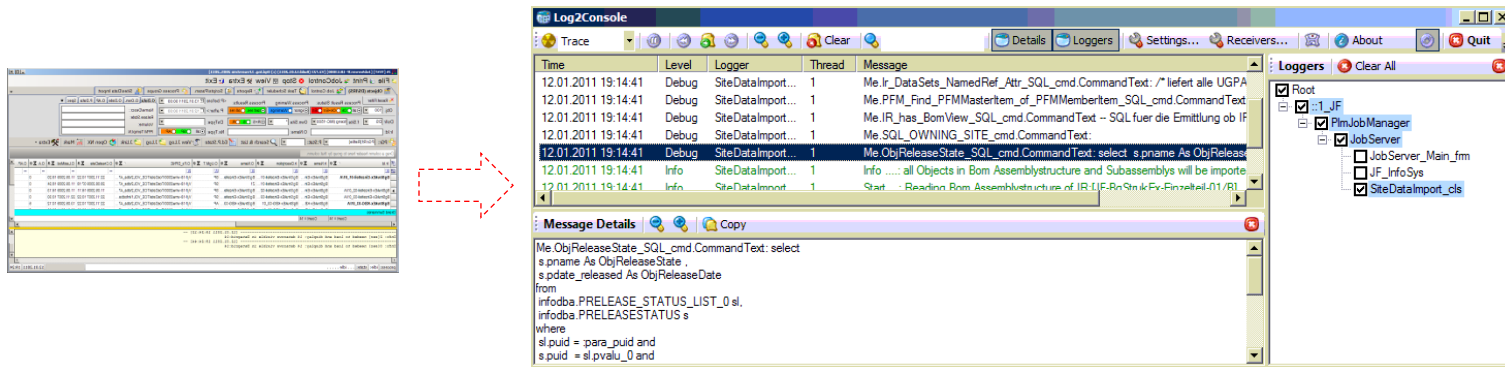
- Click Receivers (2) → Add (3) → “UDP (IP v4 and v6)” (4)
- For W7 Systems (or any System using IPv6) set “Use IPv6 Addresses” True (5)



Set on Win7  
Use IPv6  
Addresses to  
True



After setting up the software JobServer JobClient or JobPlaner is sending Log information's to Log2Console Software



# Fehlerbeseitigung Font (new 23.12.2011)

- Nach dem Start des JobServers kommt die Meldung: ‚Arial Narrow‘ does not support Style ‚Regular‘

Wenn der Start des JobManager mit der folgenden Meldung beendet wird



so muss auf diesem Rechner der Arial Narrow Font ausgetauscht werden.  
 Alle Programme beenden (nur evtl erforderlich) Systemsteuerung → Fonts



Nach dem Entfernen diesen via Drag + Drop den Schriftarten hinzufügen.

