



# J2U – die -Laufzeitumgebung für Unix

## Technische Daten



Copyright 2010 itgain Consulting GmbH

Kleine Reichenstraße 1

Im Reichenhof

D-20457 Hamburg

Tel.: +49 (0) 40 2841 755-0

Fax. +49 (0) 40 2841 755-55

Email: [j2u@itgain.de](mailto:j2u@itgain.de)

Internet: [www.itgain.de](http://www.itgain.de)



## Inhaltsverzeichnis

Was ist J2U?.....	5
Architektur von J2U .....	5
Einbindung von J2U ins Rechenzentrum.....	6
J2U auf einer dezentralen Plattform.....	6
J2U in einem z/OS-Rechenzentrum .....	7
Technische Einzelheiten von J2U.....	8
J2U-Laufzeitumgebung .....	8
J2U-Datenbank .....	8
J2U Control Panel – das Web-Interface zu J2U.....	8
J2U-Clients .....	8
Von der J2U-Runtime unterstützte JCL-Anweisungen .....	9
Voraussetzungen für J2U .....	11
Beispielaufruf mit J2U .....	12
Beispieljob J2U0025 .....	12
Starten des Jobs, z. B. von einer lokalen Konsole .....	13
Anmeldung über das J2U Control Panel.....	13
Anzeigen aller Jobs mit dem J2U Control Panel .....	14
Auswahl des gewünschten Jobs, Anzeigen der Listen des Jobs .....	14
Anzeigen der Liste JOBMSG .....	15
Weitere Produkt-Informationen .....	16



## Abbildungsverzeichnis

Abbildung 1 Architektur von J2U.....	5
Abbildung 2: Einsatz von J2U unter Unix .....	6
Abbildung 3: J2U in einem z/OS-Rechenzentrum.....	7
Abbildung 4: Re-Hosteter Sort-Job .....	12
Abbildung 5 Starten eines Jobs unter J2U.....	13
Abbildung 6: Anmeldung über das J2U Control Panel.....	13
Abbildung 7: Anzeige aller Jobs im Control Panel.....	14
Abbildung 8: Joblisten im Control Panel.....	14
Abbildung 9: Job-Messages .....	15

## Was ist J2U?

J2U® ist der JCL-Laufzeitumgebung von itgain, die es erlaubt, Jobs, die in Host-JCL geschrieben sind, ohne jede Änderung auf einem Rechner mit einem Unix-System auszuführen. J2U stellt eine vollständige Laufzeit-Umgebung für die Ausführung von Jobs unter Unix zur Verfügung.

## Architektur von J2U

Architektonisch besteht J2U aus folgenden Komponenten:

- Der J2U-Laufzeitumgebung, die als Unix-Dämon **j2ud** auf jedem Batch-Server läuft und die Funktionen eines Job Entry Subsystems wie **JES2** oder **JES3** bereitstellt.
- Dem J2U-Client, der als Unix-Startskript **sub** einen Job an J2U übergibt.
- Der zentralen J2U-Datenbank
- Dem Web-Interface J2U Control Panel, über das alle Aktionen für die Arbeit mit dem J2U-System erreichbar sind.
- 

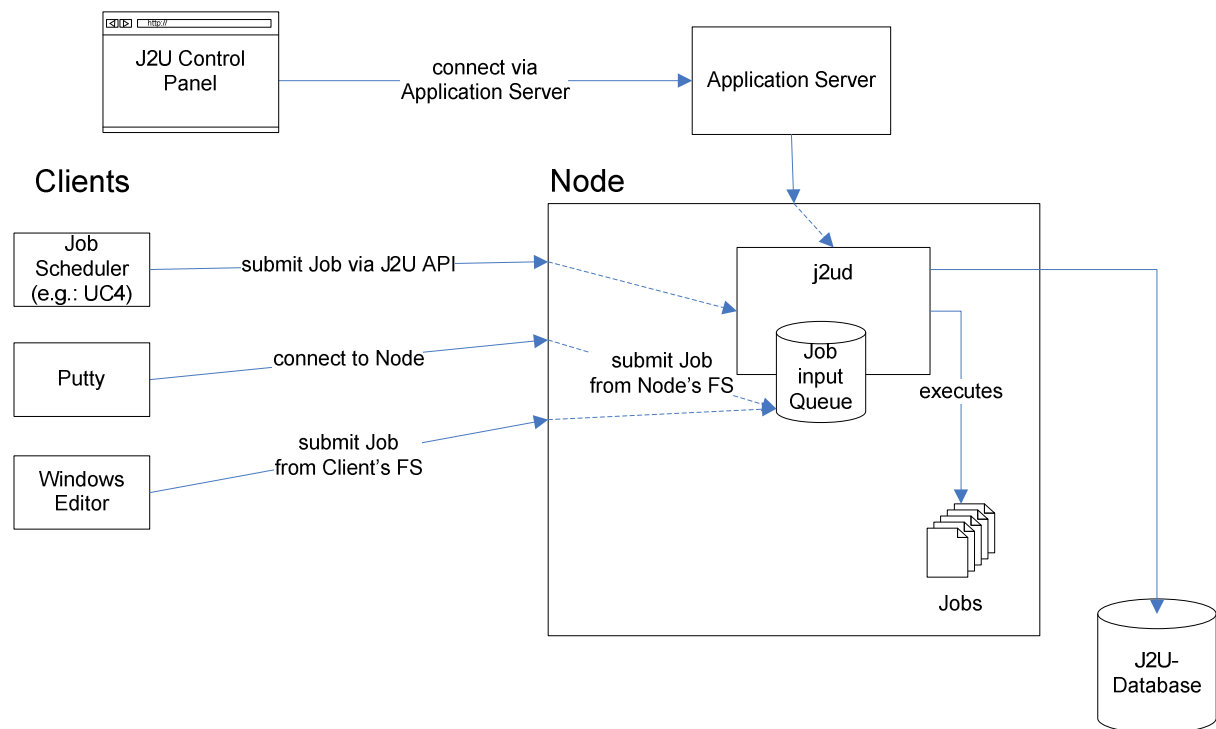


Abbildung 1 Architektur von J2U

## Einbindung von J2U ins Rechenzentrum

### J2U auf einer dezentralen Plattform

Bei der vollständigen Ablösung eines z/OS-Systems bildet J2U den Kern der Batchproduktion und übernimmt die Funktion des JESx, zusammen mit den gewohnten Utilities, dem Sort und dem Job-Controlling. An J2U sind die RZ-Produkte der dezentralen (Ziel-) Plattform angeschlossen, wie in folgender Abbildung dargestellt:

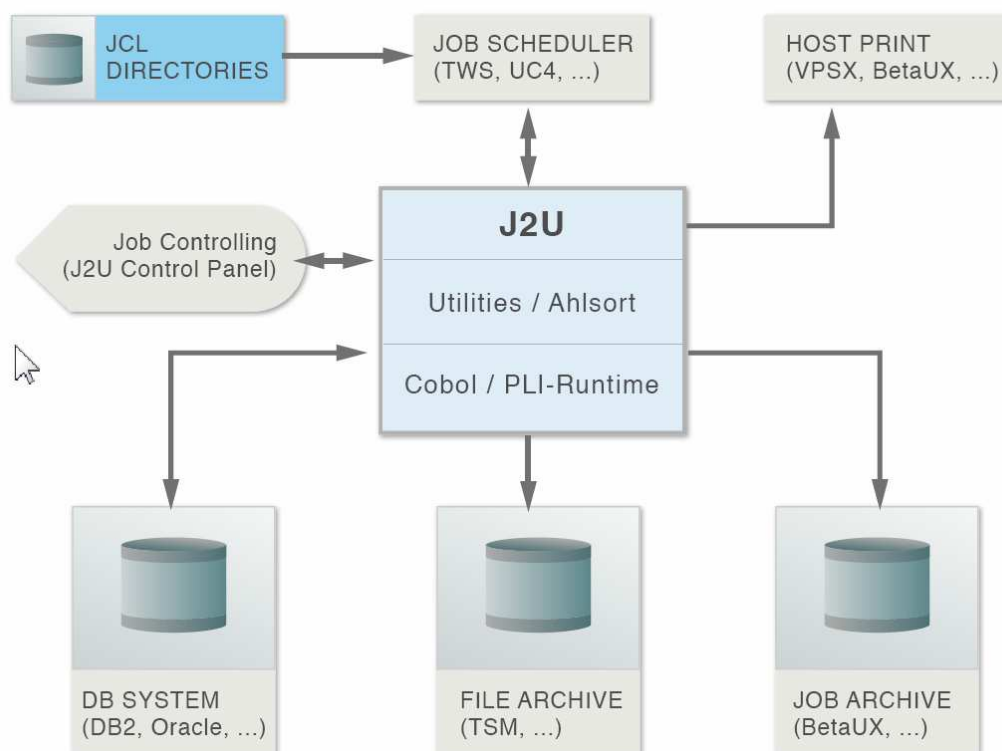


Abbildung 2: Einsatz von J2U unter Unix

Zur Zeit unterstützt J2U folgende Produkte:

- **Job-Scheduler:**  
UC4, TWS decentral, Control-M, Zeke (jeweils mit Verarbeitung von Schedulervariablen über die API des Schedulers)
- **Job-Archivierung:**  
Beta92, BetaUX
- **Drucksysteme**  
VPS, VPSX, Beta93, BetaUX
- **Datei-Archivierung und Backup**  
TSM

### J2U in einem z/OS-Rechenzentrum

J2U kann dauerhaft oder übergangsweise während eines Re-Hosting-Projekts als eigenständiger Batch-Server in den RZ-Betrieb eingebunden werden, um gezielt Batch-Jobs auf einer dezentralen (und damit kosteneffektiven) Plattform zu betreiben. Die Jobs werden dabei vom z/OS-Job-Scheduler (z. B. TWS) gesteuert, und der Output wird automatisch an die angeschlossenen RZ-Systeme weiter geleitet; Beispiel: Joblisten werden weiterhin in Beta92 archiviert, Druckdateien können weiterhin über Beta93 oder VPS verarbeitet werden.

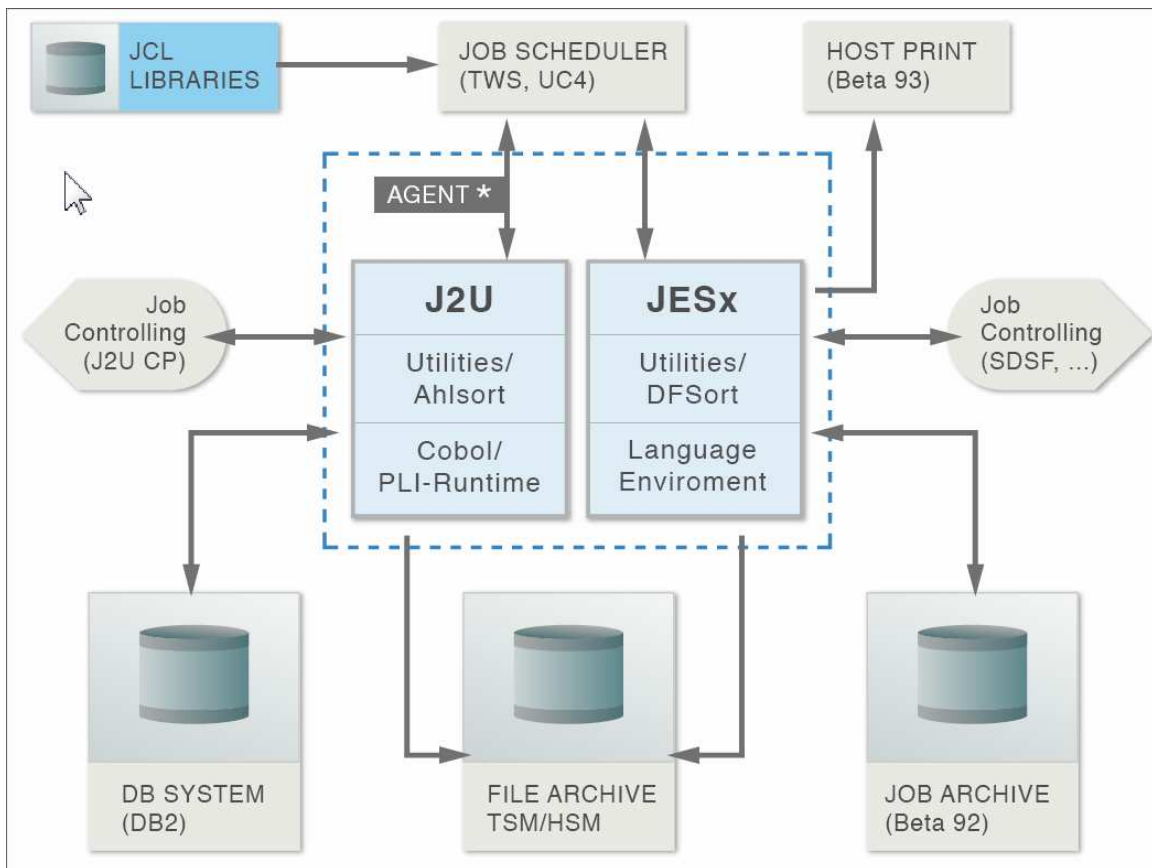


Abbildung 3: J2U in einem z/OS-Rechenzentrum

#### Hinweis:

Die Verbindung zwischen dem z/OS-Job-Scheduler und J2U übernimmt ein geeigneter Agent, z. B. Beta 48, ein TWS Agent oder ein UC4-Executor.



## Technische Einzelheiten von J2U

### ***J2U-Laufzeitumgebung***

- Ausführung von unveränderten JCL-Jobs unter Unix mit gleicher Logik wie unter z/OS
- Transparente Einbindung in Job-Scheduling-Systeme (z. B. UC4) und Editoren (Remote oder lokaler Submit)
- Transparente Integration des **DFS**ort-kompatiblen Sort-Utilites **Ahlsort** .
- Zentrales Verzeichnis aller Dateien und aller job-relevanten Informationen in RDBMS-Tabellen
- Auswerten und Berücksichtigen von Variablen verschiedener Job-Scheduler, z. B. **OPC/TWS**- und **APG-ZEKE**
- Leichte Anpassbarkeit an kundenspezifische Vorgaben (Beliebig viele Mandanten, beliebig viele Ausführungsebenen, Verhalten von Executables und ihren Laufzeitumgebungen, wie z. B. Microfocus) über XML-Konfigurationsdateien und Datenbank-Tabellen

### ***J2U-Datenbank***

- Als Datenbanksystem (unabdingbar für den Einsatz von J2U) können Sie ein bereits in Ihrem Haus eingesetztes Produkt wählen oder von itgain ein an die J2U-Lizenz gebundenes **DB2** installieren lassen, für das keine Kosten anfallen.
- J2U unterstützt zur Zeit ohne weitere Anpassungen **DB2**, **Oracle** und **Adabas-D**

### ***J2U Control Panel – das Web-Interface zu J2U***

Das browserbasierte Web-Interface *J2U Control Panel* ist das zentrale Administrations-Werkzeug für J2U; es beinhaltet die folgenden Komponenten:

- Login-Dialog J2U
- *J2U Admin* zum Administrieren von J2U selbst, z. B. Benutzerverwaltung und Konfiguration und Kontrolle von J2U-Komponenten
- *J2U Job Control* zum Kontrollieren von Jobs und Joblisten, analog zu SDSF
- *J2U File Control* zum Abfragen und Manipulieren von Dateien, analog zu den Utility-Optionen von ISPF/PDF, z. B. 3.2 Verwalten von Dateien oder 3.4 Dateiliste

### ***J2U-Clients***

Über die J2U-Clients werden Jobs auf einem J2U-Node (Batch-Server gestartet). Folgende Möglichkeiten stehen zur Verfügung:

- Lokale Anmeldung auf dem Server, z. B. über Putty oder SSH, und Starten des Jobs aus dem File System des Servers
- Remote Submit eines Jobs, der im Files System einer Windows-Workstation gespeichert ist, über einen Programm-Editor
- Starten des Jobs von einem Job-Scheduler, wie z. B. UC4, über eine spezielle API.
- Starten eines Jobs über das J2U Control Panel – File Control



### ***Merkmale von J2U***

- ▶ Interpretation von Original-Host-JCL  
Dadurch Unveränderte Geschäftslogik (Returncodes, DISP-Angaben, GDGs, Prozeduren, INCLUDE...)
- ▶ Beliebig viele Mandanten, beliebig viele Ebenen (Test, Abnahme, Produktion) pro Mandant
- ▶ Zentraler Dateikatalog, realisiert durch DB2-Tabellen
- ▶ Automatische Erkennung der Skriptsprache (JCL, bash, Perl, ...)
- ▶ Web-Interface J2U Control Panel (J2U CP)
  - ▶ Administration von J2U selbst
  - ▶ Administration der Jobs (analog SDSF,...)
  - ▶ Funktionalität der ISPF-Dataset-List (=3.4) zum komfortablen Verwalten von Dateien
- ▶ Emulation von Utilities:
  - ▶ MVS (xxxGENER, IEBCOPY, ICETOOL, IDCAMS, ...)
  - ▶ DB2 (DSNTIAUL, UNLOAD, LOAD, ...)
- ▶ Re-Hosting DFSort durch externes Sort-Utility (AHLSORT)
- ▶ SMS, HSM
  - ▶ Übernahme von SMS-Regeln (Automatische Archivierung, Aufbewahrungsfristen,..)
  - ▶ Emulieren von HSM (automatischer HRECALL) mit transparenter Einbindung von TSM
- ▶ Re-Hosting von QMF-Objekten (nur Batch)
- ▶ Unterstützung FTP, Rexx, ISPF (File Tailoring)



## **Von der J2U-Runtime unterstützte JCL-Anweisungen**

### **Hinweis:**

Alle im Folgenden nicht explizit aufgeführten Anweisungen werden von J2U ignoriert.

- ▶ **EXEC-Anweisung** mit den Parametern
  - PGM=
  - PROC=
  - PARM=
  - COND=

- ▶ **DD-Anweisung** mit den Parametern
  - DSN=
  - DISP=
  - UNIT=
  - DCB=
  - LRECL=
  - RECFM=
  - PATH=
  - SYSOUT=
  - DATACLAS=
  - LIKE=
  - SYSIN=

Unterstützt werden alle Formate der DD-Anweisung:

- **DD** [Schlüsselwort-Parameter], z. B. **DD DSN=xxx**, ...
- **DD DUMMY**
- **DD \* bzw. DD DATA DLM=** (Instream-Dateien)

J2U unterstützt außerdem

- Generation Data Groups mit den gleichen Regeln wie auf dem Host, aber mit maximal 9999 Generationen
- Rückbezüge wie z. B. **DSN=\*.STEP1.SORTOUT** oder **DCB=\*.SORTIN** sowie **Vorwärtsreferenzen** (Aufschieben von Dataset-Definitionen) durch **DD DDNAME=**
- Concatenation von beliebig vielen Dateien
- Temporäre Dateien
- Spool-Dateien (**SYSOUT=...**)

- ▶ **IF / THEN / ELSE / ENDIF-Anweisung**

- ▶ **Prozedur-Anweisungen:**

- **PROC**
- **PEND**
- **JCLLIB**

- ▶ **SET-Anweisung**



- ▶ **INCLUDE-Anweisung**
- ▶ Weitere JCL-Anweisungen:
  - `//` (Null-Anweisung)
  - `/*` (Instream-Ende-Anweisung)
  - `//*` (Kommentar-Anweisung)
- ▶ Von der J2U-Laufzeitumgebung emulierte IBM-Utilities:
  - **SORT** (über das externe Sort-Utility **Ahlsort**)
  - **IEBGENER / ICEGENER**
  - **FTP**
  - **IDCAMS** mit den Subbefehlen **DELETE**, **DEFINE/ALTER GDG**, **REPRO** und **SET MAXCC=**
  - **IEBCOPY**
  - **IEFBR14** (zum Nutzen des **DISP**-Subparameters der **DD**-Anweisung)
  - **IKJEFT01** (Zum Aufrufen von DB2- und REXX-Programmen)
- ▶ Von der J2U-Laufzeitumgebung emulierte DB2-Utilities:
  - **LOAD**
  - **UNLOAD bzw. DSNTIAUL**

### ***Voraussetzungen für J2U***

- Ein Unix-Betriebssystem; zur Zeit unterstützt J2U
  - ▶ Linux (ab Kernel 2.4)
  - ▶ AIX (Version 4.1. und höher)
  - ▶ Solaris (Version 8 und höher)
- Perl 5.8
- Ein relationales Datenbank-System, für das ein Perl-DBI- oder ein ODBC-Treiber existiert. Wenn der Kunde kein bereits vorhandenes DB-System vorgibt, installieren wir **DB2 für LUW**. Bestehende Kundeninstallationen nutzen zur Zeit **DB2 für LUW**, **Oracle** und **Adabas D**.
- Open Object Rexx (früher IBM Object Rexx), wenn Sie Rexx-Programme nutzen wollen
- Für das **J2U Control Panel**:
  - ▶ Auf Client-Seite: Ein aktueller Browser, der Javascript unterstützt,
  - ▶ Auf der Server-Seite: einen der folgenden Application Server:
    - Apache Tomcat,
    - WebSphere,
    - Sun Application Server
- Um einen Job aus einem Editor von einer Windows-Workstation zu starten:
  - ▶ Einen Datenbank-Client auf der Workstation



## Beispielaufruf mit J2U

Beispiel für einen Original-Host-Job, der unverändert unter J2U läuft. Der Job konvertiert den Inhalt einer Datei <sup>1</sup> von einem Format in ein anderes Format durch Benutzen der **OUTREC**-Anweisung des **DFSORT**. Der Job demonstriert

- Das Abbilden eines Mandanten (**mandant1**) und einer Batch-Umgebung (**j2utest**) ins Unix-Dateisystem
- Die Kompatibilität des von **J2U** benutzten Unix-Sort-Utilities **Ahlsort**
- Die Benutzung von Instream-Dateien (...**DD \***) sowie
- Rückbezüge ( **DD DSN=\*.SORT.SORTOUT**)

### Beispieljob J2U0025

```
//J2U0025 JOB (18010000),'OUTREC',
//          MSGCLASS=T,CLASS=S,
//          NOTIFY=&SYSUID,MSGLEVEL=(1,1)
//*
//RUBOUT EXEC PGM=IDCAMS
//SYSIN DD *
DELETE JO.SORTOUT
SET MAXCC = 0
//*
//***** SORT WITH OUTREC-STATEMENT
//***** Convert a file from IXF-Host format to DEL
//* (Delimited format )
//SORT EXEC PGM=SORT,REGION=6M
//SYSOUT DD SYSOUT=*
//SORTIN DD *
C05CUST_NO          NYNR49600000          000002
C09CUST_STAT        NYNR50000000          000014
C11REGION           NYNR49600000          000021
D 0010018998 00001 00000000086
D 0010018999 00001 00000000086
D 0010019000 00001 00000000086
//SORTOUT DD DISP=(NEW,PASS),LRECL=50,UNIT=WORK
//SYSIN DD *
SORT FIELDS=(08,10,A),          CUSTOMER NO
          FORMAT=CH
*
* Filter: Only Data records (file type = "D"
INCLUDE COND=(1,1,CH,EQ,C'D')
*
* Construct records in DEL-Format
* Should look like this:
* "0010019000";"00001";"00000000086"
*
OUTREC FIELDS=(01:C'""',08,10,
              12:C'";"',20,05,
              20:C'";"',27,10,
              33:C'""')
//*
//***** Show Result using backward reference
//GENER EXEC PGM=ICEGENER,COND=(4,LT)
//SYSUT1 DD DISP=(OLD,DELETE),DSN=*.SORT.SORTOUT
//SYSUT2 DD SYSOUT=*
//SYSIN DD DUMMY
```

Abbildung 4: Re-Hosteter Sort-Job

<sup>1</sup> Die Quelldatei ist durch Entladen einer DB2-Tabelle im IXF-Host-Format entstanden; die Zieldatei kann in eine DB2-Tabelle unter Linux geladen werden



## Starten des Jobs, z. B. von einer lokalen Konsole

Der Job wird über das **j2u**-Shell-Skript **sub** gestartet <sup>1</sup>

A screenshot of a PuTTY terminal window titled '192.168.10.115 - PuTTY'. The terminal shows the following text:

```
dwtest@:/data1/dr/dwtest>sub J2U0025
J2U - VERSION v3.7.12.0 - 32 bit
Copyright (C) 2004-2010 itgain consulting GmbH
J2U INFO: JOB J2U0025 (JOB 101175) SUBMITTED
J2U INFO: JOB J2U0025 (JOB 101175) ACTIVATED
J2U INFO: JOB 101175 ENDED WITH HIGHEST RC 0
dwtest@:/data1/dr/dwtest>
```

Abbildung 5 Starten eines Jobs unter J2U

## Anmeldung über das J2U Control Panel

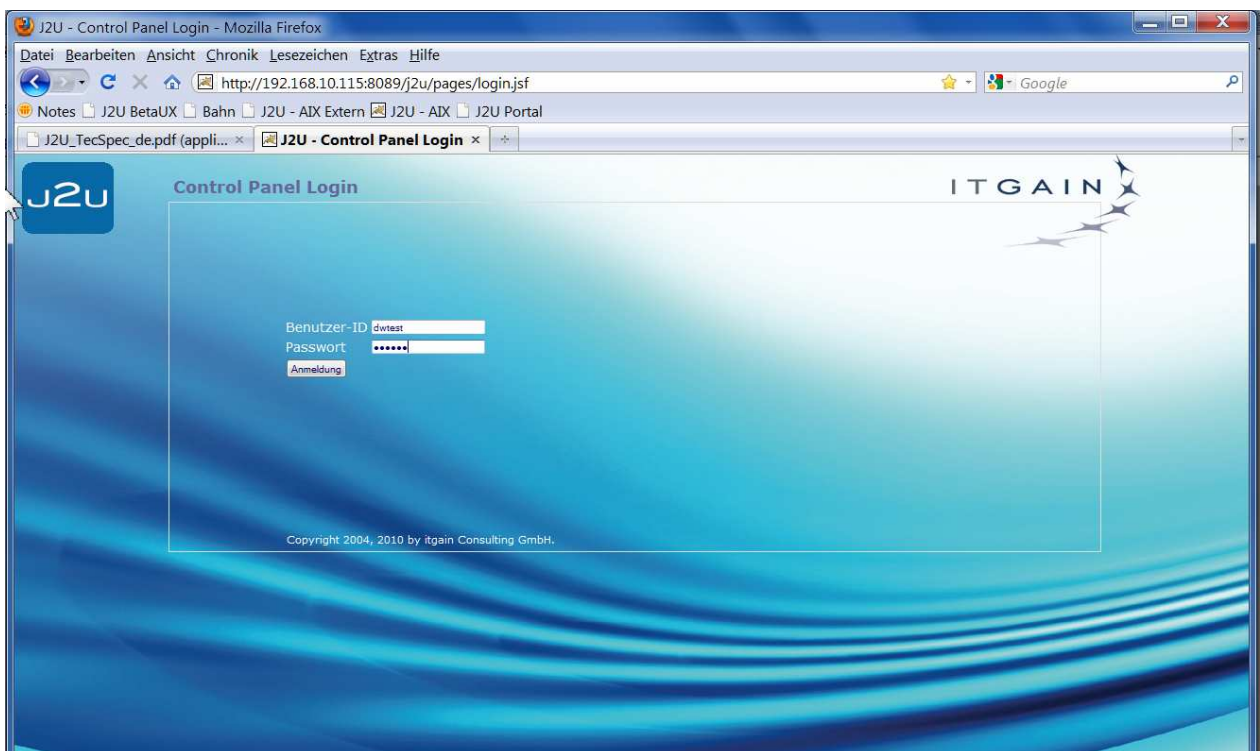


Abbildung 6: Anmeldung über das J2U Control Panel

<sup>1</sup> In Produktion wird das Shellskript durch den zuständigen Agent (Tracker, Executor) des Job-Schedulers auf dem Zielrechner gestartet.



## Anzeigen aller Jobs mit dem J2U Control Panel

Alle Listen eines Jobs (Kontroll-Listen und Spool-Output) werden in einem eigenen Verzeichnis abgelegt.

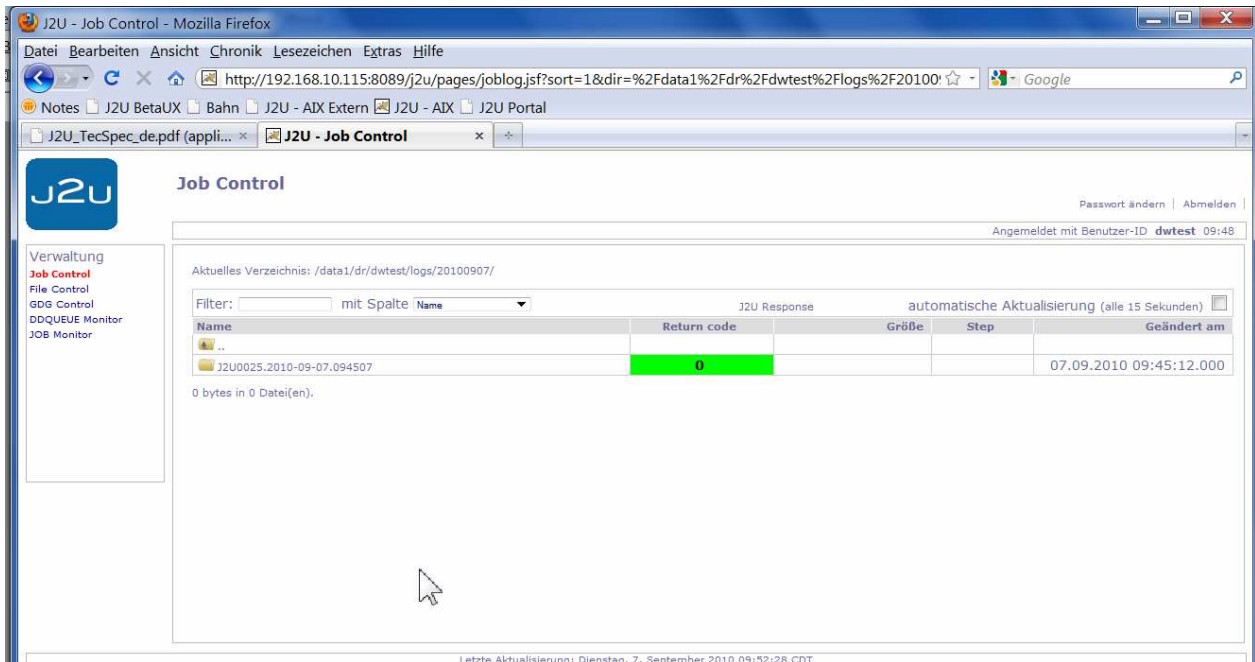


Abbildung 7: Anzeige aller Jobs im Control Panel

## Auswahl des gewünschten Jobs, Anzeigen der Listen des Jobs

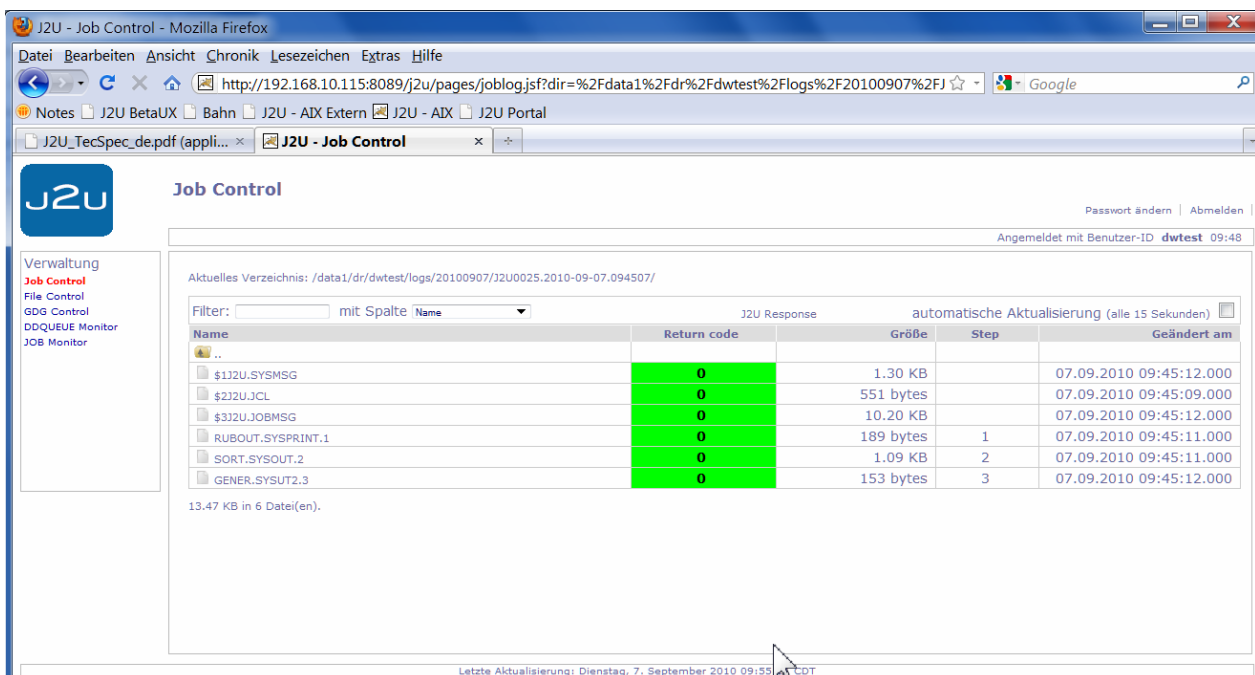


Abbildung 8: Joblisten im Control Panel



Die Dateien im Log-Verzeichnis entsprechen den auf dem Host ins JES-Spool geschriebenen Listen:

#### Hinweise:

- **\$1JES.SYSMSG** System-Messages
- **\$2JES.JCL** Aufgelöste JCL
- **\$3JOB.MSG** Jobbezogene Ausgaben
- Die übrigen Spool-Listen werden – wie auf dem Host – erzeugt durch `//xxxxx DD SYSOUT=*`

### Anzeigen der Liste **JOBMSG**

The screenshot shows the J2U Job Control web interface. The main content area displays a log file for the job `J2U0025`. The log entries include:

```
71 | 09.45.09 J2UP104I 42 // * "U010019000": "U0001": "U000000086"  
72 | 09.45.09 J2UP104I 43 //GENER EXEC PGM=ICEGENER,COND=(4,LT)  
73 | 09.45.09 J2UP104I 44 //SYSUT1 DD DISP=(OLD,DELETE),DSN=*,SORT.SORTOUT ** referback  
74 | 09.45.09 J2UP104I 45 //SYSUT2 DD SYSOUT=*  
75 | 09.45.09 J2UP104I 46 //SYSIN DD DUMMY  
76 | 09.45.09 J2UP104I 47 //SYSPRINT DD DUMMY  
77 | 09.45.09 J2UP105I LINE -----1-----2-----3-----4-----5-----6-----7-----8  
78 | 09.45.09 J2UP102I 1234567890123456789012345678901234567890123456789012345678901234567890  
79 | 09.45.09 J2UP200I END DOCUMENTATION OF JCL INPUT  
80 | 09.45.09 J2UP240I BACKWARD REFERENCE DSN=*.SORT.SORTOUT REPLACED BY DSN=___SYS-2010-09-07-094509.J2U0025.0  
81 | 09.45.09 J2US120I  
82 | 09.45.09 J2US121I  
83 | 09.45.09 J2US120I  
84 | 09.45.10 J2UR101I START J2U0025 FOR USERID dwtest  
85 | 09.45.10 J2UR102I UNIT : dr  
86 | 09.45.10 J2UR103I ENVIRONMENT: dwtest  
87 | 09.45.10 J2UR104I PATH : /data1/dr/dwtest/bin:/opt/itgain/j2u/v37/bin:/var/www/cgi-bin/CM/bin:/opt/itgain/j2u/v37/bin:/data1/dr/dwtest/data  
88 | 09.45.10 J2UR105I LIBPATH : /opt/itgain/j2u/v37/lib:/usr/lib:/lib:/home/cobra971/sqllib/lib64  
89 | 09.45.10 J2UR106I =====  
90 | 09.45.11 J2UR140I Allocations for Step J2U0025.RUBOUT  
91 | 09.45.11 J2UR120I SYSPRINT ALLOCATED TO /data1/dr/dwtest/logs/20100907/J2U0025.2010-09-07.094507/RUBOUT.SYSPRINT.1  
92 | 09.45.11 J2UR120I SYSIN ALLOCATED TO /data1/dr/dwtest/tmp/hLTsq8lyvB/INSTREAM1.3792992  
93 | 09.45.11 J2UR850I STEP J2U0025 RUBOUT ENDED WITH RC 0  
94 | 09.45.11 J2UR130I STEP TIMING INFORMATION  
95 | 09.45.11 J2UR131I 0.0100 usr 0.0100 sys + 0.0000 cusr 0.0100 csys = 0.0300 CPU  
96 | 09.45.11 J2UR106I =====  
97 | 09.45.11 J2UR140I Allocations for Step J2U0025.SORT
```

Abbildung 9: Job-Messages

#### Hinweise

- Unter anderem gibt J2U im Job-Log Aufschluss über die aktuelle Betriebssystem-Umgebung, z. B. den aktuellen Pfad und den aktuellen Library Path
- Für jeden Step werden die Allokationen aller DD-Anweisungen sowie Statistiken (CPU-Verbrauch) protokolliert
- Warnungen und Fehler werden durch das Job Control Panel farblich gekennzeichnet.



## Weitere Produkt-Informationen

<http://www.itgain.de/de/produkte/j2u/j2u.htm>