



Job Scheduling

JOB SCHEDULER

Installation und Konfiguration

August 2007

Impressum

Software- und Organisations-Service GmbH
Giesebrechtstr. 15

D-10629 Berlin

Telefon (030) 86 47 90-0

Telefax (030) 8 61 33 35

Mail info@sos-berlin.com

Web <http://www.sos-berlin.com>

Letzte Aktualisierung: Januar 2007

Inhaltsverzeichnis

1 Installation	4
1.1 Installation per Setup-Programm	4
1.2 Pakete des Setups	5
1.3 Das Paket Sample Jobs	7
1.4 Formulare des Setups	8
1.5 Verzeichnisstruktur nach der Installation	17
1.6 Automatische Installation	19
1.7 Einrichten der Datenbank	20
1.8 Starten und Stoppen des Job Schedulers	20
2 Batch-Installation	22
2.1 Konfiguration der XML-Datei	24
3 Mehrfache Installationen	27
3.1 Installation über eine bereits existierende Installation	27
3.2 Installation neben bereits existierender Installationen	27
4 Installation eines Backup Systems	28
5 Deinstallation	29
5.1 Deinstallation per Uninstaller	29
5.2 Manuelle Deinstallation unter Windows	30
5.3 Manuelle Deinstallation unter Linux/Solaris	30
6 Konfiguration	31
6.1 Datei factory.ini	31
6.2 Dateien scheduler.xml und scheduler.xsd	31
6.3 Datei jobscheduler.sh (für Unix)	31
6.4 Datei custom.inc.php	31
6.5 Konfiguration des Web Servers	33
7 Verfahren zum automatischen Update	34
7.1 Webservice SchedulerUpdate	34
7.2 Job CheckForUpdate (Client)	34
7.3 Behandlung mehrfacher Aktualisierungen	36
8 Betrieb mit 64bit	37
9 Troubleshooting	39
Index	41

1 Installation

Bei einer Neuinstallation des *Job Scheduler* sollte in folgender Reihenfolge vorgegangen werden:

- **Einrichten der Datenbank** (Seite 20) (optional)

Der *Job Scheduler* kann auch ohne Betrieb einer Datenbank genutzt werden, dann werden Job-Protokolle und Job-Historie nur im Dateisystem gespeichert. Ein weiterer Vorteil des Datenbankbetriebs ist, dass zurückliegende Protokolle mit der Web-Oberfläche des *Job Scheduler* sichtbar gemacht werden können.

Überdies gibt es Pakete im Setup, die ohne Datenbank-Unterstützung nicht ausgewählt werden können. Unterstützte Datenbanksysteme sind MySQL, Oracle, Microsoft SQL Server, PostgreSQL, Firebird und DB2.

Speziell für den Betrieb mit MySQL und MS SQL Server müssen Sie aus lizenzrechtlichen Gründen selbst einen JDBC-Treiber bereitstellen, der mit Ihrer verwendeten MySQL- bzw. MS SQL Server Version korrespondiert. Für Oracle, PostgreSQL, Firebird und DB2 wird dieser vom Setup mitgeliefert.

Beachten Sie bitte auch die Hinweise im Troubleshooting (Seite 39) zur Wahl des geeigneten JDBC-Treiber, sowie Konfiguration eines MySQL-Datenbank-Servers im ANSI-Modus

- **Installation des *Job Scheduler*** (Seite 4)

Die Installation erfolgt mittels Setup-Programm, das unter <http://www.sos-berlin.com/scheduler> heruntergeladen werden kann. Unterstützte Betriebssysteme sind Windows 2003/2003/XP, Linux ab Kernel 2.4, und Solaris 8/9/10.

- **Konfiguration des Web Servers** (Seite 33) (optional)

Der Betrieb und die Konfiguration eines Web Servers ist seitens des *Job Scheduler* nur nötig, wenn das während des Setups gewählte Paket *Web Interface* (Seite 5) genutzt werden soll.

1.1 Installation per Setup-Programm

Folgende Dateien werden unter <http://www.sos-berlin.com/scheduler> zum Herunterladen bereitgestellt.

- **scheduler_linux.tar.gz** für Linux (Archiv mit Setup-Programm)
- **scheduler_solaris.tar.gz** für Solaris (Archiv mit Setup-Programm)
- **scheduler_win32.zip** für Windows (Archiv mit Setup-Programm)
- **scheduler_jre_win32.zip** für Windows (Archiv mit Setup-Programm inklusive JRE)

Nach dem Entpacken der oben aufgeführten Archive, finden sie das Setup-Programm, das abhängig vom heruntergeladenen Archiv

- **scheduler_linux32.jar** für Linux
- **scheduler_solaris32.jar** für Solaris
- **scheduler_win32.jar** für Windows
- **scheduler_jre_win32.exe** für Windows

heißt. Die "jar"-Dateien setzen ein installiertes Java Runtime Environment voraus, während die "exe"-Datei die Java-Umgebung mitliefert.

Das Setup-Programm kann als Dialog und im Batch (siehe Batch-Installation (Seite 22)) gestartet werden.

Gestartet werden die "jar"-Dateien mittels

```
windows-shell>java -jar [download_path]\scheduler_win32.jar  
linux-shell>java -jar [download_path]/scheduler_linux32.jar  
solaris-shell>java -jar [download_path]/scheduler_solaris32.jar
```

wobei [download_path] der Speicherort der jeweiligen "jar"-Datei sein soll.
Die Datei *scheduler_jre_win32.exe* wird per Doppelklick gestartet.

Der Setup-Dialog startet, in dem Sie als erstes die Sprache des Setup wählen, nach einer Begrüßung die Lizenzbestimmungen bestätigen und den Installationspfad auswählen.

Im weiteren wird innerhalb dieser Dokumentation der Installationspfad *[install_path]* genannt. Anschließend sehen Sie den Dialog der Paketauswahl (Seite 5).

Abhängig von der Wahl der Pakete bieten weitere Formulare Konfigurationsmöglichkeiten. Näheres hierzu ist im Kapitel Formulare des Setups (Seite 8) beschrieben.

Dann werden die Dateien der gewählten Pakete kopiert und diverse Skripte ausgeführt, um die Installation ihrer Konfiguration anzupassen. Der Verlauf der vom Setup aufgerufenen Skripte wird geloggt. Sie finden die Log-Datei unter *[install_path]/logs* mit dem Namen *Install_V1.2_[datum][uhrzeit]_[laufnr].log*.

Am Ende des Setups können Sie den *Job Scheduler* bereits aufrufen, in dem Sie in Ihrem Browser (unterstützt werden Internet Explorer ab 5.5 und Firefox) als URL

```
http://localhost:[port]
```

eingeben, wobei [port] die innerhalb des Setup von Ihnen gemachte Angabe zum TCP-Port des *Job Scheduler* ist.

Für Linux/Solaris-Benutzer

Das Setup ist ein Dialog-Programm, auf dass ein X-Server installiert sein muss. Sollte kein X-Server installiert sein, so nutzen Sie bitte die Batch-Installation (Seite 22).

Der *Job Scheduler* benötigt zur Laufzeit die Bibliotheken

- [install_path]/lib/libstdc++.so.6.0.3 (Linux)
- [install_path]/lib/libstdc++.so.5.0.4 (Solaris)
- [install_path]/lib/libgcc_s.so.1

Diese sind im Setup enthalten. Es ist sicherzustellen, dass alle hierzu abhängigen Bibliotheken in der Distribution installiert sind, was z.B. mit SUSE 9 gegeben ist.

Für Windows-Benutzer

Die "jar"-Datei kann auch durch Doppelklick gestartet werden, wenn der Dateityp "jar" zum Öffnen verknüpft ist mit

```
"[Pfad zum JRE der Java-Installation]\bin\javaw.exe" -jar "%1" %*
```

1.2 Pakete des Setups

Während des Setups können folgende Pakete ausgewählt werden

- **Job Scheduler**
Das ist die Basis-Installation. Diese kann nicht abgewählt werden.
- **Update Service**
Dem *Job Scheduler* wird ein Job hinzugefügt, der wöchentlich prüft, ob es ein neues Release gibt. Hierzu muss ein Java Runtime Environment (JRE) installiert sein.
- **Database Support**
Die Job-Historie und Job-Protokolle können in einer Datenbank gespeichert werden. Unterstützte Datenbanksysteme sind MySQL, Oracle, SQL Server, PostgreSQL, DB2 und Firebird.

Abhängig von der Sprache des Setups wird die Datenbank-Tabelle `SETTINGS` gefüllt. Das kann auch nachträglich durch Aufruf des Skripts `./bin/import_settings.sh [language]` bzw. `./bin/import_settings.cmd [language]` erfolgen. Hierbei kann als `[language]` **de** für deutsch oder **en** für englisch gewählt werden.

- **Web Interface**

Die Web-Oberfläche ermöglicht die Überwachung mehrerer *Job Scheduler* und die Verwaltung von Jobs und Aufträgen. Eine Installation von PHP ab Version 4.3 wird vorausgesetzt (bei Unterstützung von Firebird ab Version 5.0).

Die Sprache des Setups bestimmt auch die Sprache der PHP-Web-Oberfläche.

- **Housekeeping Jobs**

Housekeeping Jobs werden automatisch vom *Job Scheduler* ausgeführt, bspw. für den Versand zwischengespeicherter Protokoll-eMails, das Entfernen temporärer Dateien oder den Neustart des *Job Scheduler*.

- **Sample Jobs**

Beispiele in Java, JavaScript, Perl und VBScript zum Studium für eigene Entwicklung von Scheduler Jobs.

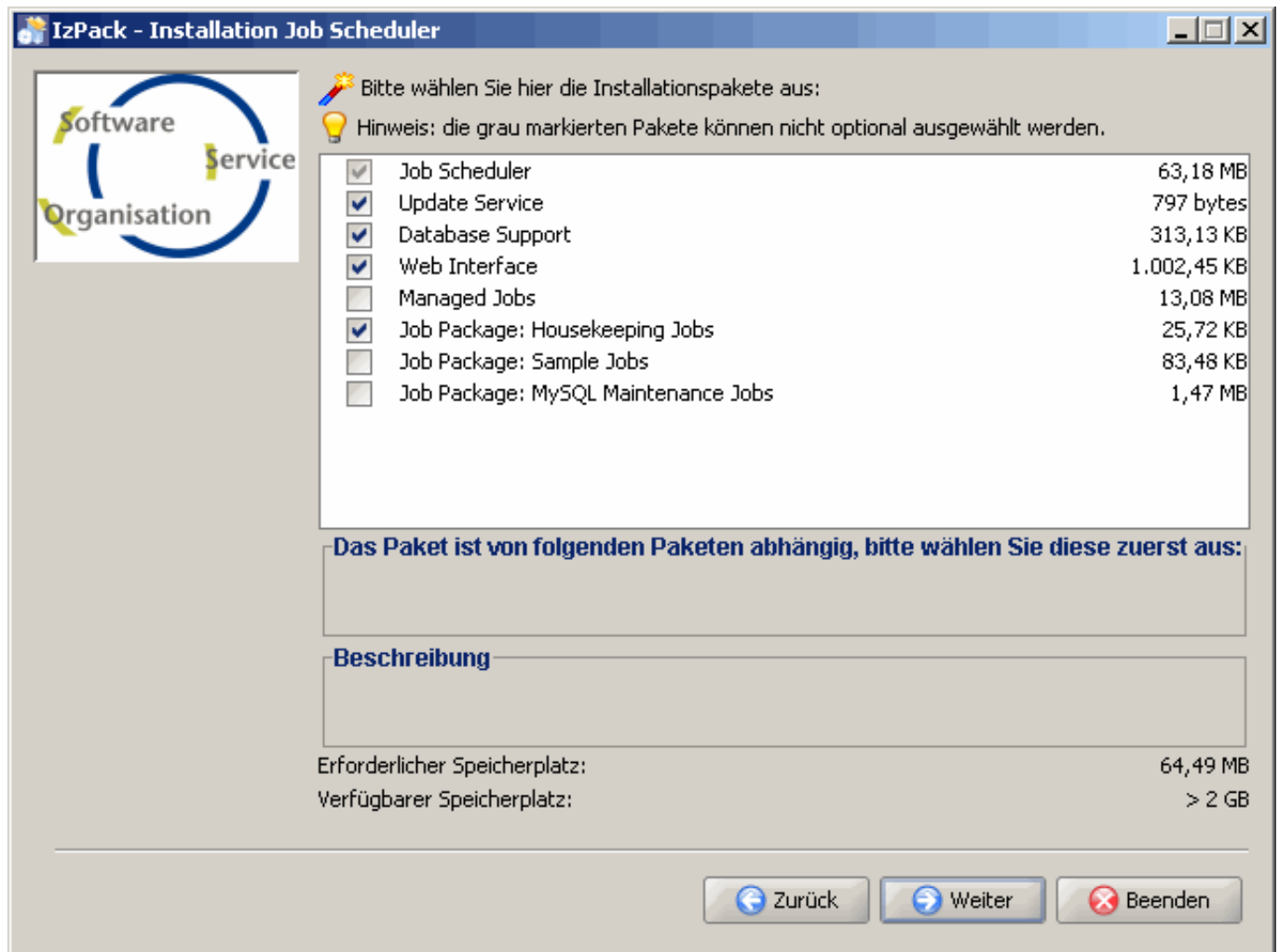
- **Managed Jobs**

Managed Jobs werden in einer Datenbank verwaltet und automatisch an einen oder mehrere *Job Scheduler* verteilt. Für den Betrieb wird eine Datenbank vorausgesetzt.

- **MySQL Maintenance Jobs**

Das Job Package für MySQL enthält Jobs für die Überwachung von Replikationen. Für den Betrieb wird eine MySQL-Datenbank vorausgesetzt.

Hierzu wird während des Setups folgender Dialog angezeigt.



1.3 Das Paket Sample Jobs

Die Beispiel-Jobs liegen nach der Installation in `[install_path]/samples`. Diese Jobs müssen dem *Job Scheduler* manuell bekannt gemacht werden. Hierzu gibt es keine Unterstützung seitens des Setups. Es muss die Datei `[install_path]/config/scheduler.xml` bearbeitet werden. Es wird unbedingt empfohlen, zuvor die Dokumentation des *Job Schedulers* zu lesen. Fehler in der Konfigurationsdatei `scheduler.xml` führen dazu, dass sich der *Job Scheduler* nicht starten lässt. Um ein Beispiel-Job hinzuzufügen, ergänzen Sie die Datei `scheduler.xml` um ein weiteres `<base>`-Element. Im folgenden Beispiel ist das die Zeile

```
<base file = "../samples/config/scheduler_sample_vbscript.xml"/>
```

Beispiel der `scheduler.xml` mit inkludierten "Sample Jobs" in VBScript:

```
<?xml version="1.0" encoding="iso-8859-1"?>
```

```
<spooler>
```

```

    <config spooler_id      = "scheduler"
            tcp_port      = "4444"
            udp_port      = "4444"
            mail_xslt_stylesheet = "config/scheduler_mail.xsl">

```

```
    <!-- included job configurations -->
```

```
<base file = "scheduler_automation_java.xml"/>
<base file = "../samples/config/scheduler_sample_vbscript.xml"/>

<!-- host name, IP address or network address of hosts, -->
<!-- that are allowed to communicate with the job scheduler -->
<security ignore_unknown_hosts = "yes">
    <allowed_host host = "localhost" level = "all"/>
</security>

<process_classes>
    <!-- max. number of processes in default process class -->
    <process_class          max_processes = "10" />
    <!-- max. number of processes running in single instances -->
    <process_class name = "single"    max_processes = "10" />
    <!-- max. number of processes running in multiple instances -->
    <process_class name = "multi"    max_processes = "10" />
</process_classes>

</config>


</spooler>
```

Anschließend ist der *Job Scheduler* (neu) zu starten. Für das Beispiel in JavaScript muss zudem ein FTP-Server und ein Java JRE installiert sein.

1.4 Formulare des Setups

Die Anzahl der während des Setups angezeigten Formulare ist abhängig von der Wahl der Pakete.

1.4.1 Formulare des Basis-Pakets Job Scheduler



IzPack - Installation Job Scheduler

Netzwerk-Konfiguration

Die Eingabe einer Scheduler ID ist sinnvoll, wenn mehrere Job Scheduler betrieben werden.

Scheduler ID:

Die Konfiguration des TCP/UDP Ports ist erforderlich, wenn der Job Scheduler mittels Web Interface überwacht werden soll, andernfalls bitte '0' angeben.

Host:

TCP Port:

UDP Port:

Es wird empfohlen für 'localhost' den TCP-Zugriff zuzulassen, optional können andere Host-Namen oder IP-Adressen angegeben werden. Um allen Servern im Netzwerk den Zugriff zu ermöglichen, geben Sie die IP-Adresse '0.0.0.0' an.

Zugelassener Rechner:

Im Eingabefeld *Scheduler ID* geben Sie bitte eine ID des *Job Scheduler* an. Verwenden Sie hierbei nur Kleinbuchstaben und/ oder Zahlen, aber keine Sonderzeichen. Unter Windows bestimmt die ID zudem den Namen des nach dem Setup installierten Dienstes in der Form *sos_scheduler_[scheduler_id]*.

Die Angabe des TCP-Ports wird für die Kommunikation mit der Web-Oberfläche benötigt.

Die *zugelassenen Rechner* genügen einem Sicherheits-Feature des *Job Scheduler*, wonach dieser die Kommunikation auf bestimmte Rechner einschränken kann. Näheres hierzu entnehmen Sie bitte der Dokumentation zum *Job Scheduler*.

Die Angaben zum Host und TCP-Port sind konfiguriert in der Datei *[install_path]/web/custom/custom.inc.php*. Die Angaben zur Scheduler ID, dem TCP-Port, dem UDP-Port und den zugelassenen Rechnern sind konfiguriert in der Datei *[install_path]/config/scheduler.xml*. Beide Dateien können nachträglich manuell geändert (Seite 31) werden.



IzPack - Installation für Job Scheduler

eMail-Konfiguration

Geben Sie bitte die Einstellungen für den Mail Server an.

IP-Adresse oder Host Ihres Mail-Servers:

Bei Ausfall des Mail-Servers können eMails mit Job-Protokoll gespeichert und von Standard-Jobs nachträglich versendet werden.

Verzeichnis zwischengespeicherter eMails: ▼

Wählen Sie bitte in welchen Fällen eMails mit Job-Protokollen automatisch versendet werden.

☐ kein eMail im Fehlerfall senden

☒ eMail im Fehlerfall senden (empfohlen)

☐ kein eMail im Fall von Warnungen senden

☒ eMail im Fall von Warnungen senden (empfohlen)

☒ kein eMail im Erfolgsfall senden (empfohlen)

☐ eMail im Erfolgsfall senden

Geben Sie hier Ihren SMTP-Server an und bestimmen Sie, wann der *Job Scheduler* Job-Protokolle automatisch versenden soll.

Diese Angaben sind konfiguriert in der Datei `[install_path]/config/factory.ini`, die nachträglich manuell geändert (Seite 31) werden kann.



IzPack - Installation für Job Scheduler

eMail-Empfänger Konfiguration

Geben Sie die Adressen der eMail-Empfänger an, die automatisch Job-Protokolle erhalten sollen. Trennen Sie mehrere Empfänger durch Komma.

Absender von Job-Protokollen:

Empfänger von eMails:

Empfänger von Kopien:

Unsichtbare Empfänger von Kopien:

Hinweis für Windows Anwender:

Wenn kein Java Runtime Environment (JRE) eingesetzt wird, müssen Sie manuell die eMail-Komponente installieren, die im Verzeichnis `install\JMail44_standard.exe` Ihres Installationsverzeichnis enthalten ist, andernfalls werden keine eMails vom Job Scheduler versendet.

Der *Job Scheduler* versendet gemäß den Einstellungen des vorherigen Formulars automatisch Job-Protokolle. Bestimmen Sie hier Absender, Empfänger und ggf. CC und BCC des Mailversands. Bei Angabe mehrere Email-Adressen müssen diese durch Komma getrennt werden.

Um den automatischen eMail-Versand des *Job Scheduler* unter Windows ohne installiertes Java Runtime Environment zu nutzen, muss das beiliegende Setup in `[install_path]\install\JMail44_standard.exe` von JMail ausgeführt werden.

Diese Angaben sind konfiguriert in der Datei `[install_path]/config/factory.ini`, die nachträglich manuell geändert (Seite 31) werden kann.

1.4.2 Formular des Pakets Update Service



IzPack - Installation Job Scheduler

Update-Service-Konfiguration

Der Job Scheduler prüft wöchentlich, ob ein neueres Release vorliegt. Wenn ja, wird Ihnen eine entsprechende eMail zugestellt. Darüber hinaus kann ein automatischer Download des Release gestartet werden. Es wird dann eine Datei namens 'scheduler_win32_update.zip' im Installationsverzeichnis des Job Schedulers gespeichert.

Wählen Sie den Wochentag: Montag

Geben Sie die Startzeit im Format HH:MM an: 20 : 00

☐ mit automatischen Download

Zurück Weiter Beenden

Dem *Job Scheduler* wird ein Job hinzugefügt, der wöchentlich prüft, ob ein neueres Release vorliegt. Diesem Job können drei Parameter zugewiesen werden. Der Wochentag und die Uhrzeit an der der Job gestartet werden soll und ob ein automatischer Download erfolgt. Im Falle des automatischen Downloads wird je nach Betriebssystem eine Datei namens *scheduler_win32_update.zip*, *scheduler_linux_update.tar.gz* oder *scheduler_solaris_update.tar.gz* im Installationsverzeichnis des *Job Schedulers* gespeichert.

Für nachträgliche Konfiguration des Jobs öffnen Sie bitte die Datei *./config/scheduler_update_service.xml*.
Weiteres zum Update-Verfahren entnehmen Sie bitte der Dokumentation *./doc/de/scheduler_update_service.pdf*.

1.4.3 Formulare des Pakets Database Support



IzPack - Installation Job Scheduler

Backup-Konfiguration

Der Job Scheduler kann unabhängig zu eventuell weiteren oder als primärer Job Scheduler eines Backup Systems installiert werden.

☒ unabhängig
☐ als primärer Job Scheduler eines Backup Systems

Wenn ein Backup System installiert werden soll, dann sollte unbedingt im letzten Dialog dieses Setup das Skript zur automatischen Installation generiert werden.

Zurück Weiter Beenden

Mit obigen Radiobuttons bestimmen Sie, ob der Scheduler standalone oder geclustert als primärer *Job Scheduler* eines Backup Systems installiert werden soll (siehe auch Installation eines Backup Systems (Seite 28)).



IzPack - Installation für Job Scheduler

Datenbank-Konfiguration

DBMS-Typ: MySQL

ACHTUNG! Sollten Sie Daten einer etwaig bestehenden Installation verändert haben, so werden diese Änderungen rückgängig gemacht. Von Ihnen hinzugefügten Daten bleiben erhalten. Bei Erstinstallation sollte die Checkbox an sein.

☒ Ja, die DB-Tabellen sollen (neu) erzeugt werden.

Geben Sie die Zugangsparameter für die Datenbank an. Die Voreinstellungen der Ports lauten für MySQL 3306, Oracle 1521, SQL-Server 1433.

Server:

Port:

Datenbank:

Benutzer:

Kennwort:

← Zurück Weiter → ✗ Beenden

In obiger Auswahlliste bestimmen Sie Ihr Datenbanksystem und in den Eingabefeldern die Datenbank-Verbindung. Ist die Checkbox in der Mitte angekreuzt (empfohlen), so ruft das Setup ein Skript auf, dass die nötigen Tabellen einrichtet und füllt. Andernfalls müssen Sie die Tabellen manuell (Seite 20) einrichten.

Diese Angaben sind konfiguriert in der Datei `[install_path]/config/factory.ini` und die `[install_path]/web/custom/custom.inc.php`. Beide Dateien können nachträglich manuell geändert (Seite 31) werden.



Wenn Ihr zuvor gewähltes Datenbanksystem weder MySQL noch MS SQL Server ist, dann überspringen Sie diesen Dialog. Das Setup ruft später ein Skript zum Einrichten der Datenbank-Tabellen auf, das den JDBC-Treiber Ihres Datenbanksystems benötigt. Für Oracle, PostgreSQL, Firebird und DB2 sind diese im Setup enthalten, während aus lizenzrechtlichen Gründen Sie hier den MySQL- bzw. MS SQL JDBC-Treiber angeben müssen, der zu Ihrer Datenbank-Version passt.

Da im weiteren auch der *Job Scheduler* einen JDBC-Treiber Ihres Datenbanksystems benötigt, wird dieser vom Setup nach `[install_path]/lib` kopiert.

Diese Angabe ist konfiguriert in der Datei `[install_path]/config/factory.ini`, die nachträglich manuell geändert (Seite 31) werden kann.

Ist das gewählte Datenbanksystem Firebird, dann muss sicher gestellt sein, dass während der Installation keine anderen Verbindungen zum Datenbank-Server existieren.

1.4.4 Formular des Pakets Housekeeping Jobs



Job-Konfiguration

Standard-Jobs zur Automation werden für folgende Aufgaben ausgeliefert: Wiederholter eMail-Versand von Protokollen nach Ausfall des Mail-Servers, Entfernen temporärer Dateien, Neustarten des Job Schedulers.

Wählen Sie eine Implementierung der Standard-Jobs.

Java (empfohlen) ▼

Zurück Weiter Beenden

Die *Housekeeping Jobs* des *Job Scheduler* sind in Java, JavaScript, VBScript und Perl implementiert. Mit der hier angebotenen Auswahlliste wählen Sie die Implementierung. Der Umfang der *Housekeeping Jobs* variiert zwischen den verschiedenen Skriptsprachen.

Sprache	Housekeeping Job
Java	scheduler_dequeue_mail, scheduler_restart, scheduler_rotate_log, scheduler_cleanup_history, scheduler_cleanup_files, scheduler_check_sanity
JavaScript	scheduler_dequeue_mail, scheduler_restart
VBScript	scheduler_dequeue_mail, scheduler_restart, scheduler_rotate_log, scheduler_cleanup_history
Perl	scheduler_dequeue_mail, scheduler_restart, scheduler_rotate_log

Dokumentationen zu diesen Jobs finden Sie nach der Installation in `[install_path]/jobs` im HTML-Format. Diese Angaben sind konfiguriert in der Datei `[install_path]/config/scheduler.xml`, die nachträglich manuell geändert (Seite 31) werden kann.

1.5 Verzeichnisstruktur nach der Installation

Die Inhalte der nachfolgend aufgeführten Verzeichnisse sind sowohl abhängig von der Wahl der Pakete während des Setups, als auch vom Betriebssystem. In diesen Fällen ist der Paketname und/ oder das Betriebssystem nach dem Verzeichnis oder der Datei in Klammern angefügt. Ist der Paketname und/ oder das Betriebssystem einem Verzeichnis angefügt, so gilt die Abhängigkeit für alle Dateien in diesem Verzeichnis.

Der *Job Scheduler* enthält einen eingebauten HTTP-Server für seine eigene Web-Oberfläche. Die hierfür nötigen Dateien befinden sich im Verzeichnis `[install_path]/config/html`. Mit dieser Web-Oberfläche ist nicht die PHP-Oberfläche gemeint, die als Paket des Setups gewählt werden kann.

Ab dem `[install_path]` des *Job Scheduler* liegt die folgende Verzeichnisstruktur vor:

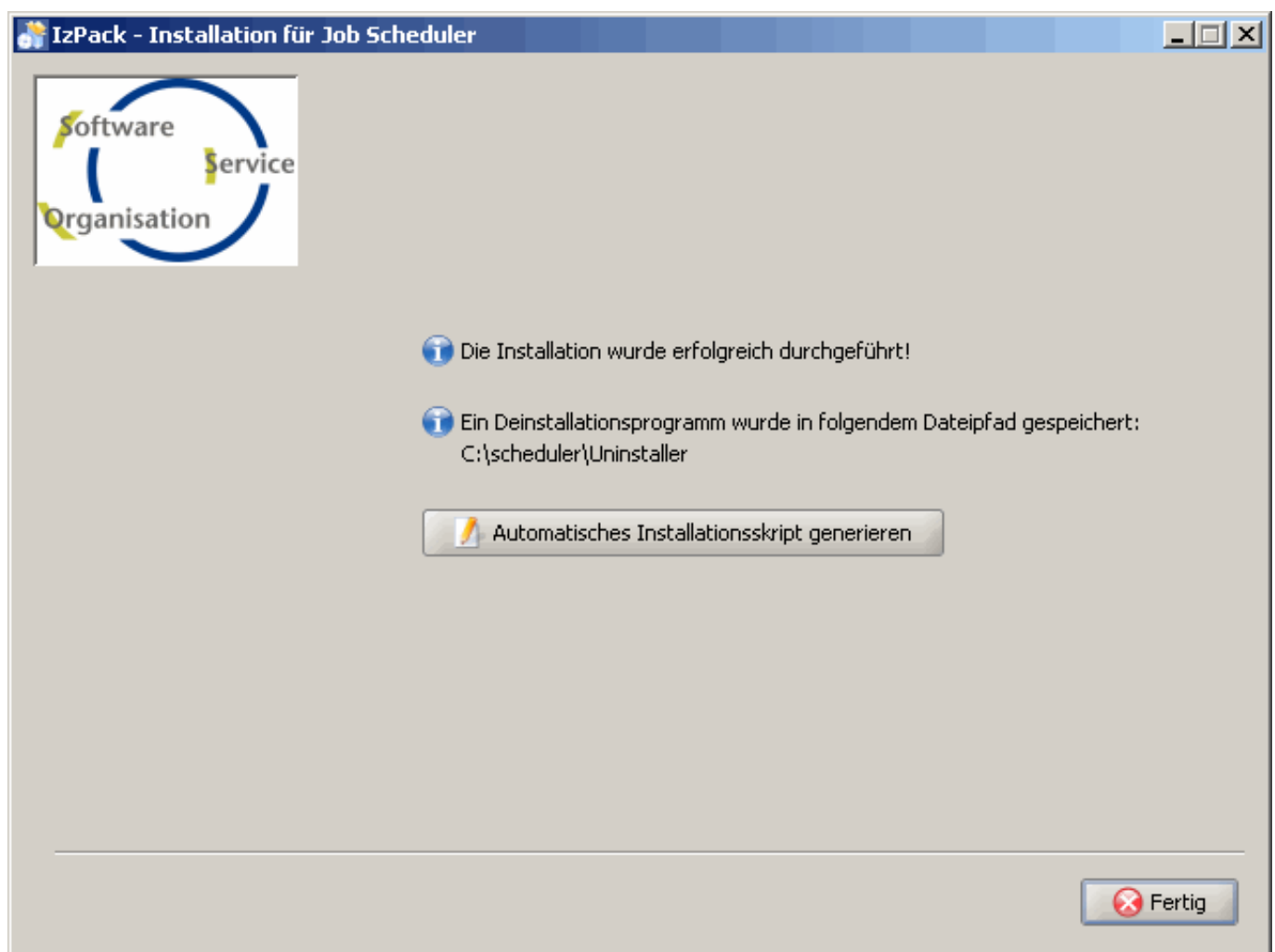
- + **bin** (Windows)
 - **hostjava.dll** Programmbibliothek
 - **hostole.dll** Programmbibliothek
 - **jobeditor.cmd** Startskript des Job Configuration Editor für XML-Konfigurationsdateien
 - **jobscheduler.cmd** Startskript des Job Schedulers
 - **managedJobChainExport.cmd** Export-Skript für Managed Jobs
 - **managedJobChainImport.cmd** Import-Skript für Managed Jobs
 - **scheduler.exe** Ausführbare Datei des *Job Scheduler*
 - **scheduler.exe.local** Dummy zur lokalen Verwendung von DLLs
 - **settingsImport.cmd** Import-Skript für Einstellungen in der Datenbank
 - **spidermonkey.dll** Programmbibliothek für Javascript (Mozilla)
- + **bin** (Linux/Solaris)
 - **jobeditor.sh** Startskript des Job Configuration Editor für XML-Konfigurationsdateien
 - **jobscheduler.sh** Startskript des Job Schedulers
 - **managedJobChainExport.sh** Export-Skript für Managed Jobs
 - **managedJobChainImport.sh** Import-Skript für Managed Jobs
 - **scheduler** Binary des *Job Scheduler*
 - **scheduler_safe.sh** Überwachungs-Skript zum automatisierten Neustart
 - **settingsImport.sh** Import-Skript für Einstellungen in der Datenbank
 - **setuid** Programm zur Ausführung von Skripten im Kontext anderer Benutzer, siehe FAQ
- + **config**
 - + **html** *Job Scheduler* eigene Web-Oberfläche
 - **factory.ini** Einstellungsdatei
 - **scheduler.xml** XML-Konfigurationsdatei des Job Schedulers
 - **scheduler.xsd** Schemadefinition für XML-Konfigurationsdateien
 - **scheduler_mail.xsl** Style Sheet zur Gestaltung von Protokoll-eMails
 - **sos.ini** Lizenz-Datei
 - **sos_settings.ini** Datenbank-Verbindung für Shell-Skripte
 - **scheduler_update_service.xml** (Update Service)
 - **scheduler_automation_java.xml** (Housekeeping Jobs)
 - **scheduler_automation_javascript.xml** (Housekeeping Jobs)
 - **scheduler_automation_perlscript.xml** (Housekeeping Jobs)
 - **scheduler_automation_vbscript.xml** (Housekeeping Jobs)
 - **scheduler_managed.xml** (Managed Jobs)
 - **default.xslt** (Managed Jobs)
 - **mail.xslt** (Managed Jobs)
 - **scheduler_mysql.xml** (MySQL Maintenance Jobs)
 - **scheduler_mysql_javascript.xml** (MySQL Maintenance Jobs)
 - **factory_mysql.ini** (MySQL Maintenance Jobs)
 - **replication_master_settings.ini** (MySQL Maintenance Jobs)
 - **replication_slave_settings.ini** (MySQL Maintenance Jobs)

- + **db** Datenbank-Objekte
 - **settings_insert_en.xml** (Managed Jobs)
 - **settings_insert_de.xml** (Managed Jobs)
 - + **msaccess** MS Access
 - **scheduler.mdb**
 - **scheduler_managed.mdb** (Managed Jobs)
 - + **mssql** MS SQL Server 2000, 2005
 - **scheduler.sql**
 - **scheduler_sanity.sql**
 - **scheduler_sanity_insert.sql**
 - **acl.sql** (Managed Jobs)
 - **acl_insert.sql** (Managed Jobs)
 - **mails.sql** (Managed Jobs)
 - **scheduler_managed.sql** (Managed Jobs)
 - **scheduler_managed_insert.sql** (Managed Jobs)
 - **settings.sql** (Managed Jobs)
 - **settings_insert.sql** (Managed Jobs)
 - **user_attributes.sql** (Managed Jobs)
 - **user_groups.sql** (Managed Jobs)
 - **user_groups_insert.sql** (Managed Jobs)
 - **user_variables.sql** (Managed Jobs)
 - **user_variables_insert.sql** (Managed Jobs)
 - **users.sql** (Managed Jobs)
 - **users_insert.sql** (Managed Jobs)
 - + **mysql** MySQL 4.1, 5.x
 - ***.sql** (siehe Verzeichnis mssql)
 - + **procedures** (MySQL Maintenance Jobs)
 - **scheduler_job_procedure.sql**
 - **scheduler_user_jobs.sql** (MySQL Maintenance Jobs)
 - + **oracle** Oracle 8.1.7, 9.2, 10g
 - ***.sql** (siehe Verzeichnis mssql)
 - + **fbsql** Firebird 1.5
 - ***.sql** (siehe Verzeichnis mssql)
 - + **pgsql** PostgreSQL 8.x
 - ***.sql** (siehe Verzeichnis mssql)
 - **sos.sql** (Managed Jobs)
 - + **db2** IBM DB2 8
 - ***.sql** (siehe Verzeichnis pgsql)
- + **doc** Dokumentationen inklusive API und Tutorial
- + **install** (Windows)
- + **jobs** Job-Skripte (nicht Java) und deren Dokumentationen (HTML)
- + **lib**
 - ***.jar** Java-Archive (für Java-Jobs)
 - **scheduler.dll** für Java-Debugging (Windows)
 - ***.so** Bibliotheken (Linux/Solaris)
- + **logs** Ablage der Log-Dateien
- + **samples** (Sample Jobs)
- + **Uninstaller** Deinstallations-Programm

- + **web** PHP-Oberfläche (Web Interface)
 - + **custom** Konfigurationsdatei für die PHP-Oberfläche
 - + **doc** Über den Web Server erreichbare Dokumentationen
 - + ... weitere Verzeichnisse

1.6 Automatische Installation

Am Ende des Setups haben Sie die Möglichkeit eine XML-Datei zur automatisierten Installation abzuspeichern, in dem durch den Knopf *Automatisches Installationsskript generieren* einen Speicher-Dialog öffnen. In ihr werden alle während des Setups gemachten Angaben gespeichert. Sollten Sie auf mehreren Rechnern dieselbe Installation ausführen wollen, so brauchen Sie nicht jedesmal die Formulare ausfüllen, sondern nutzen das Automations-Skript.



Für den Aufruf des Setups mit Automations-Skript siehe Batch-Installation (Seite 22).

1.7 Einrichten der Datenbank

Es wird empfohlen, dem *Job Scheduler* einen Datenbankbenutzer mit dazugehöriger Datenbank (bzw. Schema) zur Verfügung zu stellen. Wie Datenbanken eingerichtet werden, entnehmen Sie bitte der Dokumentation Ihres Datenbanksystems. Unterstützt werden MS SQL Server, MySQL, PostgreSQL, DB2, Firebird und Oracle Datenbanksysteme. Die Tabellen werden vom Setup erstellt, sofern während des Setups das Paket *Database Support* (Seite 5) ausgewählt und im entsprechenden Setup-Dialog die Datenbankverbindung angegeben wurde.

Des weiteren werden vom Setup in den Konfigurationsdateien `[install_path]/config/factory.ini` und `[install_path]/web/custom/custom.inc.php` die Datenbankverbindung eingetragen.

1.7.1 Manuelles Einrichten der Tabellen

Wurden die Tabellen nicht vom Setup bei der Installation des *Job Scheduler* erstellt, so können Sie dieses manuell nachholen, in dem Sie das Skript `[install_path]/install/scheduler_install_tables.(sh|cmd)` aufrufen.

Prüfen Sie, ob in den Konfigurationsdateien (Seite 31) `[install_path]/config/factory.ini`, `[install_path]/config/sos_settings.ini` und `[install_path]/web/custom/custom.inc.php` die Datenbankverbindung richtig eingetragen ist.

1.8 Starten und Stoppen des Job Schedulers

1.8.1 Job Scheduler Demon für Unix

Unter Unix wird der *Job Scheduler* als demon betrieben. Zum Starten und Stoppen verwenden Sie bitte das Skript:

```
[install_path]/bin/jobscheduler.sh start
```

```
[install_path]/bin/jobscheduler.sh stop
```

Neben start und stop verarbeitet dieses Skript weitere Parameter, z.B. debug, restart, abort und kill.

Um den *Job Scheduler* automatisch nach einem Reboot des Rechners zu starten, kopieren Sie dieses Skript bitte in das entsprechende startup/shutdown Verzeichnis, normalerweise ist dies `/etc/init.d`.

Nach der Installation wird der *Job Scheduler* nicht automatisch gestartet, verwenden Sie hierzu bitte das obige Skript.

1.8.2 Job Scheduler Dienst für Windows

Unter Windows wird der *Job Scheduler* als Dienst installiert. Sie finden den Dienst, in dem Sie in der Systemsteuerung die Management-Konsole für Dienste öffnen und nach dem Dienstenamen suchen, der mit "SOS Job Scheduler" beginnt.

Um den *Job Scheduler* manuell zu starten, stellen Sie bitte zunächst sicher, dass der Dienst nicht gestartet ist und verwenden Sie das folgende Skript:

```
[install_path]/bin/jobscheduler.cmd start
```

```
[install_path]/bin/jobscheduler.cmd stop
```

Neben start und stop verarbeitet das Skript weitere Parameter, z.B. debug, restart, cancel und kill.

Der *Job Scheduler* Dienst wird automatisch nach der Installation gestartet.

2 Batch-Installation

Wird das Setup mit einer XML-Datei parametrisiert aufgerufen, so wird kein Dialog geöffnet.

```
shell>java -jar [setup.jar] [batch_install.xml]
```

Hierbei ist [setup.jar] das für Ihr Betriebssystem anzuwendende Setup-Programm (Seite 4) und [batch_install.xml] eine bestimmte XML-Datei (siehe unten). Eine solche XML-Datei wird beispielsweise durch den Knopf *Automatisches Installationsskript generieren* des Setups erzeugt.

Die als Parameter zu verwendenden XML-Dateien müssen folgende Form haben:

```
<AutomatedInstallation langpack="eng">
  <com.izforge.izpack.panels.HelloPanel/>
  <com.izforge.izpack.panels.InfoPanel/>
  <com.izforge.izpack.panels.LicencePanel/>
  <com.izforge.izpack.panels.TargetPanel>
    <installpath>./scheduler</installpath>
  </com.izforge.izpack.panels.TargetPanel>
  <com.izforge.izpack.panels.PacksPanel>
    <selected>
      <!-- SELECT THE PACKS YOU WANT TO INSTALL -->
      <pack index="0"/>
      <pack index="1"/>
      <pack index="2"/>
      <pack index="3"/>
      <!--pack index="4"/-->
      <pack index="5"/>
      <!--pack index="6"/-->
      <!--pack index="7"/-->
    </selected>
  </com.izforge.izpack.panels.PacksPanel>
  <com.izforge.izpack.panels.UserInputPanel>
    <userInput>
      <!-- Network Configuration -->
      <entry key="serviceHost" value="localhost"/>
      <entry key="serviceId" value="scheduler"/>
      <entry key="serviceAllowedHost" value="localhost"/>
      <entry key="serviceTcpPort" value="4444"/>
      <entry key="serviceUdpPort" value="4444"/>
    </userInput>
  </com.izforge.izpack.panels.UserInputPanel>
  <com.izforge.izpack.panels.UserInputPanel>
    <userInput>
      <!-- Backup Configuration -->
      <entry key="clusterOptions" value=""/>
    </userInput>
  </com.izforge.izpack.panels.UserInputPanel>
  <com.izforge.izpack.panels.UserInputPanel>
    <userInput>
      <!-- Mail Configuration -->
      <entry key="mailServer" value="localhost"/>
      <entry key="mailQueueDirectory" value=""/>
      <entry key="mailOnWarning" value="yes"/>
      <entry key="mailOnSuccess" value="no"/>
      <entry key="mailOnError" value="yes"/>
    </userInput>
  </com.izforge.izpack.panels.UserInputPanel>
</AutomatedInstallation>
```

```

    </userInput>
</com.izforge.izpack.panels.UserInputPanel>
<com.izforge.izpack.panels.UserInputPanel>
    <userInput>
        <!-- Mail Recipients Configuration -->
        <entry key="mailFrom" value="scheduler@localhost"/>
        <entry key="mailTo" value=""/>
        <entry key="mailBcc" value=""/>
        <entry key="mailCc" value=""/>
    </userInput>
</com.izforge.izpack.panels.UserInputPanel>
<com.izforge.izpack.panels.UserInputPanel>
    <userInput>
        <!-- Update Configuration -->
        <entry key="checkForUpdateStarttime" value="20:00"/>
        <entry key="checkForUpdateStartday" value="1"/>
        <entry key="autoUpdateDownload" value="0"/>
    </userInput>
</com.izforge.izpack.panels.UserInputPanel>
<com.izforge.izpack.panels.UserInputPanel>
    <userInput>
        <!-- Housekeeping Job Configuration -->
        <entry key="jobAutomationLanguage"
            value="scheduler_automation_java.xml"/>
    </userInput>
</com.izforge.izpack.panels.UserInputPanel>
<com.izforge.izpack.panels.UserInputPanel>
    <userInput>
        <!-- Database Configuration -->
        <entry key="databaseDbms" value=""/>
        <entry key="databaseCreate" value="off"/>
        <entry key="databaseHost" value="localhost"/>
        <entry key="databasePort" value=""/>
        <entry key="databaseSchema" value="scheduler"/>
        <entry key="databaseUser" value="scheduler"/>
        <entry key="databasePassword" value="scheduler"/>
    </userInput>
</com.izforge.izpack.panels.UserInputPanel>
<com.izforge.izpack.panels.UserInputPanel>
    <userInput>
        <!-- Configuration for JDBC Driver -->
        <entry key="connector" value=""/>
    </userInput>
</com.izforge.izpack.panels.UserInputPanel>
<com.izforge.izpack.panels.UserInputPanel>
    <userInput/>
</com.izforge.izpack.panels.UserInputPanel>
<com.izforge.izpack.panels.InstallPanel/>
<com.izforge.izpack.panels.ProcessPanel/>
<com.izforge.izpack.panels.FinishPanel/>
</AutomatedInstallation>

```

Diese XML-Datei spiegelt alle Werte wider, die während eines Setup-Dialogs angegeben werden können.

Eine Beispiel-XML liegt dem Installations-Archiv bei und heißt *scheduler_install.xml*.

2.1 Konfiguration der XML-Datei

Konfigurierbar sind das Attribut langpack im Root-Element <AutomatedInstallation>, das <installpath>-Element, die <pack>-Elemente und die Attribute value in den <entry>-Elementen.

Element/Attribut	Beschreibung
langpack	Hier legen Sie die Sprache der Anwendung fest. Mögliche Werte sind 'eng' für englisch und 'deu' für deutsch.
installpath	Hier ist der Installationspfad anzugeben
selected	<p>Das <selected>-Element besitzt Elemente der Form <pack index="[integer]"/>. Für jedes existierende <pack>-Element wird ein Paket des Setups gewählt.</p> <ul style="list-style-type: none"> • <pack index="0"/> ist das <i>Job Scheduler</i> Basis Paket. Dieses Paket muss immer angegeben sein. • <pack index="1"/> ist das Paket <i>Update Service</i>. • <pack index="2"/> ist das Paket <i>Database Support</i>. • <pack index="3"/> ist das Paket <i>Web Interface</i>. • <pack index="4"/> ist das Paket <i>Managed Jobs</i>. Für dieses Paket muss <pack index="2"/> und <pack index="3"/> angegeben sein. • <pack index="5"/> ist das Paket <i>Housekeeping Jobs</i>. • <pack index="6"/> ist das Paket <i>Sample Jobs</i>. • <pack index="7"/> ist das Paket <i>MySQL Maintenance Jobs</i>. Für dieses Paket muss <pack index="2"/> angegeben sein. <p>Informationen zu den Paketen und deren etwaigen Voraussetzungen entnehmen Sie bitte dem Kapitel Pakete des Setups (Seite 5). Bitte beachten Sie unbedingt die obig beschriebenen Abhängigkeiten der Pakete untereinander.</p>
Netzwerk-Konfiguration	
value von key="serviceHost"	Geben Sie den Namen oder die IP-Adresse des Hosts an, auf dem der <i>Job Scheduler</i> betrieben wird.
value von key="serviceld"	Die Eingabe einer Scheduler ID ist sinnvoll, wenn mehrere <i>Job Scheduler</i> betrieben werden.
value von key="serviceAllowedHost"	Es wird empfohlen für 'localhost' den TCP-Zugriff zuzulassen, optional können andere Host-Namen oder IP-Adressen angegeben werden. Um allen Servern im Netzwerk den Zugriff zu ermöglichen, geben Sie die IP-Adresse '0.0.0.0' an.
value von key="serviceTcpPort"	Die Konfiguration des TCP/UDP Ports ist erforderlich, wenn der <i>Job Scheduler</i> mittels Web Interface überwacht werden soll, andernfalls bitte '0' angeben.
value von key="serviceUdpPort"	siehe serviceTcpPort
Backup-Konfiguration	

Element/Attribut	Beschreibung
value von key= "clusterOptions"	Der <i>Job Scheduler</i> kann unabhängig zu eventuell weiteren, als primärer <i>Job Scheduler</i> eines Backup Systems oder als Backup Scheduler eines Backup Systems installiert werden. Ein Backup System kann nur mit einer Datenbank betrieben werden. Ohne Datenbank oder als unabhängiger <i>Job Scheduler</i> muss der Wert leer bleiben. Als primärer <i>Job Scheduler</i> ist der Wert '-exclusive' anzugeben. Als Backup <i>Job Scheduler</i> ist der Wert '-exclusive -backup -backup-precedence=[N]' anzugeben, wobei [N] für eine ganze Zahl steht (siehe Installation eines Backup Systems (Seite 28)).
Email-Konfiguration	
value von key= "mailServer"	IP-Adresse oder Host Ihres SMTP-Servers
value von key= "mailQueueDirectory"	Bei Ausfall des Mail-Servers können Emails mit Job-Protokoll gespeichert und von Standard-Jobs nachträglich versendet werden. Hierzu geben Sie hier das Verzeichnis zwischengespeicherter Emails an.
value von key= "mailOnWarning"	Wählen Sie hier, ob im Fall von Warnungen Emails mit Job-Protokollen automatisch versendet werden. Mögliche Werte sind 'yes' und 'no'.
value von key= "mailOnSuccess"	Wählen Sie hier, ob im Erfolgsfall Emails mit Job-Protokollen automatisch versendet werden. Mögliche Werte sind 'yes' und 'no'.
value von key= "mailOnError"	Wählen Sie hier, ob im Fehlerfall Emails mit Job-Protokollen automatisch versendet werden. Mögliche Werte sind 'yes' und 'no'.
Email-Empfänger Konfiguration	
value von key= "mailFrom"	Geben Sie den Absender von Job-Protokollen an
value von key= "mailTo"	Geben Sie die Email-Adressen der Empfänger der Job-Protokolle an. Trennen Sie mehrere Empfänger durch Komma.
value von key= "mailCc"	Geben Sie die Email-Adressen der Empfänger von Kopien der Job-Protokolle an. Trennen Sie mehrere Empfänger durch Komma.
value von key= "mailBcc"	Geben Sie die Email-Adressen der unsichtbaren Empfänger von Kopien der Job-Protokolle an. Trennen Sie mehrere Empfänger durch Komma.
Update-Service-Konfiguration Folgende Werte werden nur ausgewertet, wenn das <code><pack index="1"/></code> -Element existiert.	
value von key="check ForUpdateStarttime"	Der <i>Job Scheduler</i> prüft wöchentlich, ob ein neueres Release vorliegt. Geben Sie die Startzeit im Format HH:MM an.
value von key="check ForUpdateStartday"	Der <i>Job Scheduler</i> prüft wöchentlich, ob ein neueres Release vorliegt. Wählen Sie den Wochentag als Ziffer an (0=Sonntag, 1=Montag, ..., 6=Samstag).
value von key= "autoUpdateDownload"	Wenn ein neueres Release vorliegt, wird Ihnen eine entsprechende eMail zugestellt. Darüber hinaus kann ein automatischer Download des Release gestartet werden. Mögliche Werte sind 1 (mit automatischen Download) oder 0 (ohne automatischen Download).

Element/Attribut	Beschreibung
Job-Konfiguration Folgender Wert wird nur ausgewertet, wenn das <code><pack index="5"/></code> -Element existiert	
value von key="jobAutomationLanguage"	Standard-Jobs zur Automation werden für folgende Aufgaben ausgeliefert: Wiederholter eMail-Versand von Protokollen nach Ausfall des Mail-Servers, Entfernen temporärer Dateien, Neustarten des <i>Job Schedulers</i> . Wählen Sie eine Implementierung der Standard-Jobs. Mögliche Werte sind 'scheduler_automation_java.xml' (empfohlen), 'scheduler_automation_javascript.xml', 'scheduler_automation_vbscript.xml' und 'scheduler_automation_perl.xml'.
Datenbank-Konfiguration Folgende Werte werden nur ausgewertet, wenn das <code><pack index="2"/></code> -Element existiert	
value von key="databaseDbms"	Wählen Sie ein Datenbanksystem. Mögliche Werte sind 'mysql' (für MySQL), 'oracle' (für Oracle), 'mssql' (für MS SQL Server), 'pgsql' (für PostgreSQL), 'fbsql' (für Firebird) und 'db2' (für DB2).
value von key="databaseCreate"	Entscheiden Sie, ob die Datenbank-Tabellen (neu) erzeugt werden sollen. Mögliche Werte sind 'on' (mit Erzeugen) und 'off' (ohne Erzeugen).
value von key="databaseHost"	Geben Sie den Host-Namen oder die IP-Adresse des Datenbankrechners an.
value von key="databasePort"	Geben Sie den Port der Datenbankinstanz an. Die Voreinstellungen der Ports lauten für MySQL 3306, Oracle 1521, SQL-Server 1433, PostgreSQL 5432, Firebird 3050, DB2 50000.
value von key="databaseSchema"	Geben Sie den Namen der Datenbank an.
value von key="databaseUser"	Geben Sie den Benutzernamen für den Datenbankzugriff an.
value von key="databasePassword"	Geben Sie das Kennwort für den Datenbankzugriff an.
Konfiguration für JDBC Treiber Folgender Wert wird nur ausgewertet, wenn <code><pack index="2"/></code> -Element existiert	
value von key="connector"	Sollten Sie einen anderen DBMS-Typ als MySQL oder MS SQL Server gewählt haben, so können Sie diesen Wert leer lassen. Haben Sie MySQL oder MS SQL Server als DBMS-Typ gewählt, müssen Sie den entsprechenden JDBC Treiber bereitstellen. Aus lizenzrechtlichen Gründen ist kein JDBC Treiber für MySQL oder MS SQL Server im Installationsprogramm verfügbar. Bitte geben Sie hier den Pfad des JDBC Treiber an (z.B. mysql-connector-java-*.jar für MySQL, sqljdbc.jar für MS SQL Server).

3 Mehrfache Installationen

3.1 Installation über eine bereits existierende Installation

Hierunter wird eine Installation verstanden, deren Installationsverzeichnis dasselbe ist, wie eine bereits auf dem gleichen Rechner existierende Installation des *Job Scheduler*.

Beim Überinstallieren werden nicht alle angezeigten Formulare des Setups zur Konfiguration (vollständig) ausgewertet.

Bis auf die Angabe der Scheduler ID können Sie die drei Formulare des Basis-Pakets *Job Scheduler* (Seite 9) überspringen. Wenn Sie Änderungen der *Netzwerk-* und/ oder *Email-Konfiguration* wünschen, so müssen Sie diese manuell konfigurieren (Seite 31).

Das Formular *Datenbank-Konfiguration* (Seite 13) muss selbst dann erneut ausgefüllt werden, wenn sich nichts an der Datenbank-Verbindung verändert hat.

Wenn die Checkbox für das Einrichten der Tabellen angekreuzt ist, dann wird vom Setup wiederum ein Skript aufgerufen, das jetzt nur Datenbank-Tabellen erzeugt, die es noch nicht gibt. Es werden jedoch alle Datensätze der Erstinstallation auf ihre ursprünglichen Werte zurückgesetzt. Daten, die nach der Erstinstallation hinzugefügt wurden, bleiben unberührt.

Im Formular *Konfiguration für den JDBC Treiber* muss der JDBC Treiber **nicht** angegeben werden, wenn ihr Datenbanksystem unverändert ist.

Änderungen im Formular *Housekeeping Jobs* (Seite 16) werden berücksichtigt.

3.2 Installation neben bereits existierender Installationen

Hierunter wird eine Installation verstanden, deren Installationsverzeichnis noch nicht von einer anderen auf dem gleichen Rechner bereits existierenden Installation des *Job Scheduler* verwendet wird.

Folgendes ist im Formular *Netzwerk-Konfiguration* (Seite 9) des Basis-Pakets *Job Scheduler* unbedingt zu beachten

- Die *Scheduler ID* ist über alle installierten *Job Scheduler* des Rechners eindeutig zu wählen. Unter Windows bestimmt die ID des *Job Scheduler* zudem den Namen des nach dem Setup installierten Dienstes in der Form *sos_scheduler_[scheduler_id]*. Dienstenamen müssen pro Rechner eindeutig sein.
- Der *TCP-Port* ist über alle installierten *Job Scheduler* des Rechners eindeutig zu wählen.

Es wird empfohlen allen installierten *Job Scheduler* die gleiche Datenbank-Verbindung zuzuweisen. Insbesondere bei Verwendung des Pakets *Managed Jobs* ist dies zwingend.

Das Paket *Web Interface* braucht nicht erneut installiert werden, wenn die Datenbank-Verbindung unverändert zugewiesen wird. Vielmehr bietet es sich dann an, in der Datei *[install_path]/config/scheduler.xml* (Seite 31) einen *main_scheduler* zu definieren. Host und Port dieses *main_scheduler* sind in die Datei *[install_path]/web/custom/custom.inc.php* (Seite 31) manuell einzutragen.

4 Installation eines Backup Systems

Zuerst installieren Sie einen primären *Job Scheduler* für das Backup System wie im Kapitel Installation (Seite 4) beschrieben. Wählen Sie während des Setups das Paket *Database Support*, da ein Backup System nur mit Unterstützung einer Datenbank betrieben werden kann. Im Dialog *Backup-Konfiguration* wählen Sie die Installation als primärer *Job Scheduler* eines Backup Systems aus. Im letzten Dialog des Setup sollten Sie unbedingt das automatische Installationsskript generieren.

Die Installation des Backup *Job Schedulers* erfolgt dann per Batch-Installation (Seite 22). Öffnen sie hierzu vorher das automatische Installationsskript in einem Texteditor und passen Sie die Werte im Element '<installpath>' und im Attribut 'value' des <entry>-Elements mit 'key="clusterOptions"' an.

Geben Sie für die 'clusterOptions' den Wert '-exclusive -backup -backup-precedence=[N]' an, wobei [N] für eine ganze Zahl steht.

Wenn mehrere Backup *Job Scheduler* einen ausgefallenen primären *Job Scheduler* (-exclusive) ersetzen können, übernimmt der mit dem niedrigsten Wert von -backup-precedence. Wenn Sie -backup-precedence=[N] weglassen, wird der Initialwert '1' genommen (0 ist für den primären *Job Scheduler* reserviert). Des weiteren setzen Sie den Wert des Attributs 'value' im <entry>-Elements mit 'key="databaseCreate"' auf 'off', da bereits bei der Installation der primären *Job Scheduler* die Datenbank eingerichtet sein müsste.

Gegebenenfalls sind noch weitere Werte wie 'serviceHost' anzupassen, jedoch müssen die Angaben zur Datenbank und zur Scheduler ID unverändert bleiben, da alle *Job Scheduler* eines Backup Systems die gleiche Datenbank-Verbindung und gleiche Scheduler ID haben müssen.

Unter Windows wird ein Dienst des Backup *Job Schedulers* mit dem Namen `sos_scheduler_[scheduler_id]_backup` eingerichtet.

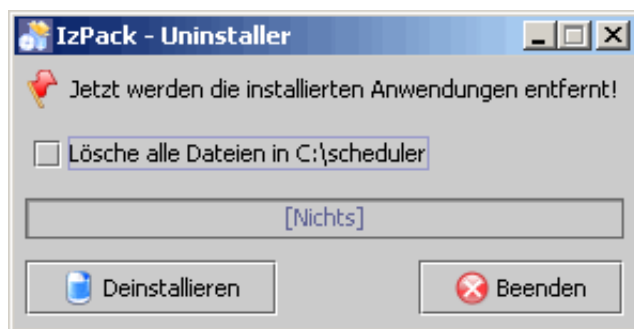
5 Deinstallation

5.1 Deinstallation per Uninstaller

Der für die Installation ausgeführte Setup installiert mithin das Deinstallations-Programm `[install_path]/Uninstaller/uninstaller.jar`. Gestartet wird die Deinstallation mittels

```
windows-shell>java -jar [install_path]\Uninstaller\uninstaller.jar  
unix-shell>java -jar [install_path]/Uninstaller/uninstaller.jar
```

anschließend fordert Sie ein Dialog zur Bestätigung der Deinstallation auf.



Eine ggf. eingerichtete Datenbank für den *Job Scheduler* müssen Sie manuell löschen. Des weiteren sind ggf. virtuelle Verzeichnisse im Web Server manuell zu löschen.

Für Linux/Solaris-Benutzer Der Uninstaller ist ein Dialog-Programm, daher muss ein X-Server verfügbar sein.

Für Windows-Benutzer

Das Deinstallations-Programm kann auch durch Doppelklick gestartet werden, wenn der Dateityp "jar" zum Öffnen verknüpft ist.

```
"[Pfad zum JRE der Java-Installation]\bin\javaw.exe" -jar "%1" %*
```

Wenn IIS als Web Server für die *Job Scheduler* Web-Oberfläche konfiguriert wurde, sind entsprechende virtuelle Verzeichnisse vor der Deinstallation zu löschen, sonst können entsprechende physische Verzeichnisse vom Uninstaller nicht restlos entfernt werden.

Der installierte Dienst "SOS Job Scheduler id=[scheduler_id]" sollte nach der Deinstallation entfernt sein. Hierbei ist [scheduler_id], die von Ihnen bei der Installation gewählte Id des *Job Scheduler*. Mitunter ist dieser aber nur als *deaktiviert* gekennzeichnet. Dann ist der Dienst erst nach einem Neustart des Rechners entfernt. Sie können dies prüfen, in dem Sie den Dienste-Dialog öffnen (Start->Ausführen services.msc) oder die Kommandozeile

```
C:\>net start sos_scheduler_[scheduler_id]
```

eingeben. Je nach Zustand des Dienstes wird der Aufruf den Text *"Der angegebenen Dienst kann nicht gestartet werden. Er ist deaktiviert oder nicht mit aktivierten Geräten verbunden."* oder *"Der Name für den Dienst ist ungültig."* liefern. Sollte der Dienst nur deaktiviert sein, so ist eine erneute Installation eines *Job Scheduler* mit gleicher [scheduler_id] erst nach einem Neustart möglich.

5.2 Manuelle Deinstallation unter Windows

Für die manuelle Deinstallation öffnen Sie eine Shell (Start->Ausführen cmd) und führen dann folgende Schritte aus. Hierbei ist [install_path] der Pfad zum Installationsverzeichnis des *Job Scheduler*.

- **Web Server rekonfigurieren**
Wenn ein Web Server für die Web-Oberfläche des *Job Scheduler* konfiguriert wurde, dann sind entsprechende virtuelle Verzeichnisse zu entfernen. Speziell bei Verwendung des IIS können sonst nicht alle Verzeichnisse restlos entfernt werden.
- **Job Scheduler beenden**
C:\>[install_path]\bin\jobscheduler.cmd stop
Wenn der *Job Scheduler* bereits beendet war, wird eine Fehlermeldung erscheinen, die ignoriert werden kann.
- **Dienst des Job Scheduler deinstallieren**
C:\>[install_path]\bin\jobscheduler.cmd remove
- **Datenbank entfernen**
Zum Entfernen einer etwaigen für den *Job Scheduler* eingerichteten Datenbank lesen Sie bitte die Dokumentation ihres Datenbanksystems.
- **Programmbibliothek hostole.dll deregistrieren**
C:\>regsvr32 \u [install_path]\bin\hostole.dll
- **Alle Dateien und Verzeichnisse löschen**
C:\>rmdir /S /Q [install_path]

5.3 Manuelle Deinstallation unter Linux/Solaris

Für die manuelle Deinstallation öffnen Sie eine Shell und führen dann folgende Schritte aus. Hierbei ist [install_path] der Pfad zum Installationsverzeichnis des *Job Scheduler*.

- **Web Server rekonfigurieren**
Wenn ein Web Server für die Web-Oberfläche des *Job Scheduler* konfiguriert wurde, dann sind entsprechende virtuelle Verzeichnisse zu entfernen.
- **Job Scheduler beenden**
shell>[install_path]/bin/jobscheduler.sh stop
Wenn der *Job Scheduler* bereits beendet war, wird eine Fehlermeldung erscheinen, die ignoriert werden kann.
- **Datenbank entfernen**
Zum Entfernen einer etwaigen für den *Job Scheduler* eingerichteten Datenbank lesen Sie die Dokumentation ihres Datenbanksystems.
- **Alle Dateien und Verzeichnisse löschen**
shell>rm -r -f [install_path]

6 Konfiguration

Die Konfiguration des *Job Scheduler* erfolgt über die Konfigurationsdateien:

- **factory.ini**
- **scheduler.xml**
- **custom.inc.php** (konfiguriert die Web-Oberfläche)
- **jobscheduler.sh** (für Unix)

Diese werden vom Setup entsprechend Ihrer Angaben konfiguriert.

6.1 Datei factory.ini

Die Datei *factory.ini* befindet sich im Verzeichnis *[install_path]/config*. In ihr werden insbesondere Email-Einstellungen, Datenbankverbindung und Klassenpfad der java-Archive gesetzt. Näheres hierzu ist in *Job Scheduler* Dokumentation beschrieben.

6.2 Dateien scheduler.xml und scheduler.xsd

Die Dateien *scheduler.xml* und *scheduler.xsd* befinden sich im Verzeichnis *[install_path]/config*. In der Datei *scheduler.xml* werden u.a. Host und Port des *Job Scheduler*, sowie Jobs mit deren Laufzeiten, Job-Ketten und Prozessklassen bestimmt. Näheres hierzu ist in *Job Scheduler* Dokumentation beschrieben.

Die Datei *scheduler.xsd* enthält das XML Schema der Konfiguration, gegen das die Datei *scheduler.xml* validiert wird.

Zur Verwaltung von Job-Konfigurationen in der Datei *scheduler.xml* wird ein graphischer Editor ausgeliefert. Sie starten den Editor unter Windows wie folgt:

```
[install_path]\bin\jobeditor.cmd
```

Unter Unix benötigen Sie einen X-Server und starten den Editor mit dem Skript:

```
[install_path]/bin/jobeditor.sh
```

Der Editor validiert automatisch Konfigurations-Dateien gegen das Schema *scheduler.xsd*.

6.3 Datei jobscheduler.sh (für Unix)

Die Datei *jobscheduler.sh* ist nur für Unix relevant und befindet sich im Verzeichnis *[install_path]/bin*. In ihr wird insbesondere der *LD_LIBRARY_PATH* gesetzt. Wenn der *Job Scheduler* die Java-Laufzeitumgebung nicht findet, dann muss diese hier entsprechend angepasst werden.

6.4 Datei custom.inc.php

Die Datei *custom.inc.php* befindet sich im Verzeichnis *[install_path]/web/custom*, wenn während des Setups das Paket *Web Interface* (Seite 5) gewählt wurde. Durch diese Datei wird die Web-Oberfläche konfiguriert. Es sind Datenbanktyp, Datenbankverbindungsdaten, Sprache, Host und Port des *Job Scheduler* und Timeout für *Job Scheduler*-Kommandos per TCP einstellbar.

Sprache

Es wird Englisch und Deutsch unterstützt. Zum Einstellen der Sprache ist die PHP-Konstante `SOS_LANG` mit kleingeschriebenen zweibuchstabigen Länderkennzeichen wie folgt zu setzen. Wenn der Eintrag fehlt oder leer ist, wird Deutsch verwendet.

- Für Englisch:

```
if(!defined('SOS_LANG')) { define ( 'SOS_LANG', 'en' ); }
```
- Für Deutsch:

```
if(!defined('SOS_LANG')) { define ( 'SOS_LANG', 'de' ); }
```

Datenbankverbindung

Zum Einstellen der Datenbankverbindung ist die PHP-Konstante `APP_CONNECTION_AUTH` in der Form

```
if(!defined('APP_CONNECTION_AUTH')) { define ( 'APP_CONNECTION_AUTH',  
'-db=[Datenbankname] -user=[Benutzername] -password=[Passwort]  
-host=[Servername oder -IP]:[port]' ); }
```

zu setzen. Wenn der Wert für '[port]' fehlt, dann wird der Standardport des jeweiligen Datenbanksystems verwendet. Wenn der Wert für '-host' fehlt, dann wird 'localhost' und Standardport benutzt.

Datenbanktyp

Es werden Oracle, MySQL, Microsoft SQL Server, PostgreSQL, DB2, Firebird und ODBC-Datenquellen unterstützt. Zum Einstellen des Datenbanktyps ist die PHP-Konstante `APP_CONNECTION_CLASS` wie folgt zu setzen.

- Für Oracle:

```
if(!defined('APP_CONNECTION_CLASS')) {  
define ( 'APP_CONNECTION_CLASS', 'sos_oracle_record_connection' ); }
```
- Für MySQL:

```
if(!defined('APP_CONNECTION_CLASS')) {  
define ( 'APP_CONNECTION_CLASS', 'sos_mysql_record_connection' ); }
```
- Für Microsoft SQL Server:

```
if(!defined('APP_CONNECTION_CLASS')) {  
define ( 'APP_CONNECTION_CLASS', 'sos_mssql_record_connection' ); }
```
- Für PostgreSQL:

```
if(!defined('APP_CONNECTION_CLASS')) {  
define ( 'APP_CONNECTION_CLASS', 'sos_pgsql_record_connection' ); }
```
- Für DB2:

```
if(!defined('APP_CONNECTION_CLASS')) {  
define ( 'APP_CONNECTION_CLASS', 'sos_db2_record_connection' ); }
```
- Für Firebird:

```
if(!defined('APP_CONNECTION_CLASS')) {  
define ( 'APP_CONNECTION_CLASS', 'sos_fbsql_record_connection' ); }
```
- Für ODBC-Datenquellen:

```
if(!defined('APP_CONNECTION_CLASS')) {  
define ( 'APP_CONNECTION_CLASS', 'sos_odbc_record_connection' ); }
```

Host des überwachenden Job Scheduler

```
if(!defined('APP_SCHEDULER_HOST')) { define ( 'APP_SCHEDULER_HOST', 'localhost' ); }
```


TCP-Port des überwachenden Job Scheduler

```
if(!defined('APP_SCHEDULER_PORT')) { define ( 'APP_SCHEDULER_PORT', '4444' ); }
```

Timeout

Die Web-Oberfläche schickt Kommandos an den *Job Scheduler* per TCP. Wenn dieser nicht in der hier angegebene Zeit (in Sekunden) antwortet, bricht die Web-Oberfläche die TCP-Verbindung ab.

```
if(!defined('APP_SCHEDULER_TIMEOUT')) { define ( 'APP_SCHEDULER_TIMEOUT', '5' ); }
```

6.5 Konfiguration des Web Servers

Wenn während des Setups das Paket *Web Interface* (Seite 5) gewählt wurde, wird ein PHP-fähiger Web Server benötigt mit PHP ab Version 4.3. Dieser sollte so konfiguriert werden, dass die Verzeichnisse *[install_path]/web* und *[install_path]/logs* zu erreichen sind, wobei das virtuelle Verzeichnis von *[install_path]/logs* unterhalb des virtuellen Verzeichnisse von *[install_path]/web* zeigen muss. Näheres zum Einrichten virtueller Verzeichnisse entnehmen Sie bitte der Dokumentation Ihres Web Servers.

Beispiel für Apache (*httpd.conf*):

```
Alias /scheduler/logs/ [install_path]/logs/  
Alias /scheduler/ [install_path]/web/
```

In der PHP Konfigurationsdatei *php.ini* müssen folgende Module aktiviert werden:

- *php_domxml* (je nach PHP Version bereits implementiert)
- *php_oci8* (bei Verwendung einer Oracle Datenbank)
- *php_pgsql* (bei Verwendung einer PostgreSQL Datenbank)
- *php_mssql* (bei Verwendung einer MS SQL Server Datenbank)
- *php_mysql* (bei Verwendung einer MySQL Datenbank, je nach PHP Version bereits implementiert)
- *php_ibm_db2* (bei Verwendung einer DB2 Datenbank)
- *php_interbase* (bei Verwendung einer Firebird Datenbank)

Nach Änderungen der PHP Konfigurationsdatei *php.ini* ist der Web Server neu zu starten.

7 Verfahren zum automatischen Update

Auf <http://www.sos-berlin.com> ist ein Webservice installiert, der Anfragen zur aktuellen verfügbaren Version (Release) des Job Schedulers entgegennimmt. Liegt eine neuere Version vor, wird diese Information dem Aufrufer zurückgeliefert.

In der Konfiguration des Job Schedulers wird ein Job ausgeliefert, der einmal wöchentlich eine Anfrage an den Webservice bezüglich neuerer Versionen macht. Liegt eine neue Version vor, wird eine entsprechende Email an den Systemadministrator versendet. Wahlweise kann der Job automatisch die erforderlichen Dateien herunterladen.

Installationsvoraussetzung

Der Job benötigt ein Java Runtime Environment (JRE).

7.1 Webservice SchedulerUpdate

Der Webservice läuft 24h am Tag

- Nimmt "CheckRequest" entgegen
- Ermittelt, ob neuere Version verfügbar ist.
- Schickt Antwort an den Aufrufer

Aufbau der Anforderung:

```
<CheckForUpdateRequest>
  <hostname>Hostname des Clients</hostname>
  <release>Release-Nr. aus ./config/.version</release>
  <os>Betriebssystem aus Systemabfrage per Java</os>
  <os_install>Betriebssystem aus ./config/.version </os_install>
  <product>scheduler</product>
  <automatic_download>[ 0| 1]</automatic_download>
</CheckForUpdateRequest>
```

Aufbau der Antwort:

```
<CheckForUpdateAnswer>
  <release>Wert aus Request</release>
  <new_release>aktuell vorhandene Release-Nr.</new_release>
  <os>Wert aus Request</os>
  <os_install>Wert aus Request</os_install>
  <automatic_download>Wert aus Request</automatic_download>
  <update_needed>1, wenn eine Aktualisierung vorliegt</update_needed>
</CheckForUpdateAnswer>
```

7.2 Job CheckForUpdate (Client)

Arbeitsweise:

- Startet voreingestellt einmal wöchentlich
- Schickt einen CheckRequest an den Webservice (synchron)
- Angaben über aktuelles Release wird der Datei *./config/.version* entnommen
- Email-Adresse wird aus der Datei *./config/factory.ini* entnommen
- Verbindungs-Timeout = 30 Sekunden

Jobdefinition

Die Implementierung ist im Java-Archiv `sos.scheduler.jar` enthalten.

```
<job name="check_for_update" title="Automatisches Updateverfahren für den Scheduler">
  <script java_class="sos.scheduler.job.JobSchedulerCheckUpdates" language="java"/>
  <run_time let_run="no">
    <weekdays>
      <day day="1">
        <period single_start="20:00"/>
      </day>
    </weekdays>
  </run_time>
</job>
```

File config/.version Structure:

```
[ scheduler]
release=x. x. x. x
os_install=[ windows| linux| solaris]
```

Folgende Aufrufparameter können angegeben werden:

```
webserviceUrl:      http://www.sos-berlin.com/check_for_update
product:            scheduler
ftp_host:           www.sos-berlin.com
ftp_port:           21
ftp_user:           anonymous
ftp_password:
ftp_transfer_mode:  binary
ftp_passive_mode:   1
ftp_remote_dir
ftp_local_dir
ftp_automatic_download: 0
```

Der Job schickt eine Anfrage an den Webservice und wertet die Antwort aus. Der Timeout für den Verbindungsaufbau ist fest auf 30 Sekunden eingestellt.

Wenn der Webservice meldet, dass ein neueres Release vorliegt, wird eine entsprechende Email an den Systemadministrator versendet:

```
There is a new version of the job scheduler available.
Your version is: 1.2.3.3
New version is: 1.2.7
The update-file has been downloaded
to: /home/sos/scheduler/scheduler_linux_update.tar.gz
```

Wenn eingestellt ist, dass der Job mögliche Aktualisierung automatisch herunterladen soll, wird ein entsprechender FTP-Transfer gestartet.

Im Fehlerfall schreibt der Job eine Warnung in das Protokoll.

Aufbau des Request

Der Name des Betriebssystems wird aus `System.getProperties()` und `"os.name"` ermittelt. Der Name des Hosts ist `read from InetAddress.getLocalHost` mit `getHostName()`.

Eine typische Anfrage:

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
<soapenv:Header>
<wsa:To xmlns:wsa="http://schemas.xmlsoap.org/ws/2004/08/addressing">
http://localhost:4455/check_for_update</wsa:To>
<wsa:ReplyTo xmlns:wsa="http://schemas.xmlsoap.org/ws/2004/08/addressing">
<wsa:Address>http://schemas.xmlsoap.org/ws/2004/08/addressing/role/anonymous<
/ws:Address>
</wsa:ReplyTo>
</soapenv:Header>

<soapenv:Body>
<addOrder xmlns="http://www.sos-berlin.com/scheduler">
<jobchain>check_for_update</jobchain>
<title>Scheduler: Check for Update</title>
<xml_payload>
<CheckForUpdateRequest>
<hostname>myHost</hostname>
<release>1.2.3.4</release>
<os>Windows XP</os>
<os_install>windows</os_install>
<product>scheduler</product>
<automatic_download>1</automatic_download>
</CheckForUpdateRequest>
</xml_payload>
</addOrder>
</soapenv:Body>
</soapenv:Envelope>
```

7.3 Behandlung mehrfacher Aktualisierungen

Wenn ein neues Release gefunden wurde, wird das in der Datei `./config/version` vermerkt. Wenn die Datei heruntergeladen wurde, wird der Ablageort in die Datei `./config/version` geschrieben. Nach einer Anfrage, bei der ein neues Release gemeldet wurde, sieht die Datei also wie folgt aus:

```
[ scheduler]
release=1.2.3.8
new_release=1.2.9
os_install=windows
downloaded_file=filename
```

Wenn das neue Release nicht installiert wird und der Update-Check erneut läuft, werden keine Emails mehr über ein neues Release an den Administrator geschickt. Wenn die Datei schon erfolgreich heruntergeladen wurde, werden keine Dateien mehr angefordert.

Stattdessen wird ins Protokoll geschrieben:

```
"You already have downloaded new release, but it has not been installed"
"You have already received an e-mail about a new release, but this new release has
not been installed.
See file filename"
```

8 Betrieb mit 64bit

Der Job Scheduler kann auf den unterstützten Betriebssystemen mit 64bit Betrieb eingesetzt werden. Bitte, beachten Sie die folgenden Hinweise und Voraussetzungen:

- **Job Scheduler ist ein 32bit Programm**
Der Job Scheduler ist als 32bit Programm entwickelt worden. Sie könnten jetzt fragen, weshalb die Software nicht für 64bit portiert wird? Die Antwort ist einfach: es gibt keinen Bedarf für diese Portierung.
 - Sie können die ausführbare 32bit Datei des Job Schedulers auf jeder der unterstützten 64bit Plattformen ablaufen lassen (Windows, Linux, Solaris, HP-UX PA-RISC, HP-UX Itanium). Der Betrieb des Job Schedulers erfolgt unabhängig von der Tatsache, dass Sie vermutlich 64bit Programme innerhalb Ihrer Job-Implementierungen ablaufen lassen wollen, da der Job Scheduler plattformspezifische Prozesse startet.
 - Der mögliche Gewinn an Ausführungsgeschwindigkeit zwischen einer 32bit Version und einer 64bit Version des Job Schedulers ist vernachlässigbar. Der bei weitem höhere Anteil der Ausführungszeit wird durch Jobs verbraucht, nicht durch den Job Scheduler.
 - Für eine 64bit Version des Job Schedulers ist ein forking des Codes erforderlich. Aktuell stellen die 32bit Plattformen (Windows) den größten Anteil der Downloads dar und müssen weiter unterstützt werden. Wir würden daher zusätzliche Entwicklerkapazitäten für die Pflege zweier Versionen über einen längeren Zeitraum aufwenden müssen, was effektiv die Roadmap des Job Schedulers verlangsamen würde. Wir schätzen den Nutzen neuer Features höher ein als den Nutzen aus einer vereinfachten Installation durch einen nativen 64bit Job Scheduler.
- **Java 32bit JRE**
Ein Java 32bit JRE wird für die Ausführung des Setups und für den Betrieb des Job Schedulers benötigt. Unserer Erfahrung nach ist es keine schlechte Idee ein von SUN bereitgestelltes JRE zu verwenden, da JREs anderer Distributionen, z.B. IBM Websphere, in manchen Punkten weniger vollständig bzw. für das jeweilige Produkt in einer Weise abgeändert sind, die keine ausreichende Kompatibilität mit Java Standards gewährleistet.
 - **Windows 64bit**
Sie können das im entsprechenden Job Scheduler Download enthaltene JRE 1.5 verwenden oder ein separates 32bit JRE installieren. Sie sollten das neu installierte JRE auf keinen Fall permanent in Ihre *PATH* Systemvariable aufnehmen (oder eine ggf. vorhandene *JDK JAVA_HOME* Systemvariable ändern), da dies das Verhalten anderer Programme beeinflussen würde, die auf den Betrieb mit einem 64bit JRE angewiesen sind. Die Installation können Sie in folgenden Schritten vornehmen:
 1. Führen Sie Download und Installation des 32bit JRE durch.
 2. Öffnen Sie ein Konsolenfenster und fügen Sie den Pfad des gerade installierten JRE Ihrer *PATH* Variablen hinzu, z.B.:
`C:\>SET PATH=C:\Program Files (x86)\Java\jre1.5.0_01\bin;%PATH%`
 3. Entpacken Sie das Archiv des Job Scheduler Downloads in ein temporäres Verzeichnis, z.B. `C:\temp\scheduler.`
 4. Starten Sie im Konsolenfenster Java mit dem Job Scheduler Setup aus dem temporären Verzeichnis, z.B.:
`C:\>java -jar C:\temp\scheduler\scheduler_win32.jar`
 5. Nach Abschluss des Setups fügen Sie bitte den Pfad der Java Virtual Machine Ihrer Job Scheduler Konfigurationsdatei `.\config\sos.ini` hinzu, z.B.:
[java]
vm = C:/Program Files (x86)/Java/jre1.5.0_01/bin/client/jvm.dll
 - **Unix/Linux 64bit**
Sie müssen ein 32bit JRE per Download beziehen, falls keines auf dem Rechner verfügbar ist.
 1. Führen Sie Download und Installation des 32bit JRE durch.
 2. Fügen Sie den Pfad des gerade installierten JRE Ihrer *PATH* Variablen hinzu, z.B.:
`PATH=$HOME/jdk1.5.0_06/jre/bin:$PATH`

3. Entpacken Sie das Archiv des Job Scheduler Downloads in ein temporäres Verzeichnis, z.B.

```
/tmp/scheduler.md /tmp/scheduler
cd /tmp/scheduler
tar xzf scheduler_linux.1.3.1.tar.gz .
```
4. Starten Sie Java mit dem Job Scheduler Setup aus dem temporären Verzeichnis, z.B.:

```
java -jar /tmp/scheduler/scheduler_linux32.jar
```
5. Nach Abschluss des Setups fügen Sie bitte den Pfad der Java Virtual Machine der Umgebungsvariable `LD_LIBRARY_PATH` im Startup Script `./bin/jobscheduler.sh` hinzu, z.B.:

```
LD_LIBRARY_PATH=$SCHEDULER_HOME/lib:/usr/local/lib:
$HOME/jdk1.5.0_06/jre/lib/i386:$HOME/jdk1.5.0_06/jre/lib/i386/client:$LD
_LIBRARY_PATH
```

- **Job Scheduler Setup für 64bit Plattform**

Stellen Sie bitte sicher, dass bei Ausführung des Job Scheduler Setups ein 32bit JRE verwendet wird. In Abhängigkeit von der Zielplattform kann es Unterschiede in der Installation des Job Schedulers geben:

- **Windows 64bit**

Wie jedes andere 32bit Programm müssen Sie den Job Scheduler und das Java JRE im Verzeichnis `C:\Program Files (x86)` (bzw. dem sprachspezifischen Verzeichnisnamen) für 32bit Programme installieren. Eine Installation in andere Verzeichnisse wird nicht funktionieren.

Falls dieses Verzeichnis nicht der ideale Aufbewahrungsort für Ihre Job Scheduler Konfigurations- und Protokolldateien ist, dann können Sie die Pfade für diese Dateien im Script `./bin/jobscheduler.cmd` anpassen, indem Sie die entsprechenden Kommandozeilenoptionen ändern (`-config`, `-log-dir`, `-log` etc.). Damit die Änderungen für den Betrieb des Job Schedulers als Windows-Dienst wirksam werden, sollten Sie die folgenden Kommandos verwenden (das Beispiel verwendet `C:\Program Files (x86)\scheduler` als Installationsverzeichnis):

```
C:\>cd "C:\Program Files (x86)\scheduler\bin"
C:\Program Files (x86)\scheduler\bin>jobscheduler.cmd remove
C:\Program Files (x86)\scheduler\bin>jobscheduler.cmd install
```

- **Unix/Linux 64bit**

Es gibt keinen Unterschied bzgl. der Installation des 32bit Job Schedulers für diese Plattform. Allerdings müssen Sie ggf. Änderungen am Startup Script bzgl. des JRE Installationsverzeichnisses vornehmen (siehe oben).

9 Troubleshooting

Zusätzlich zu diesen Informationen finden Sie weitere Hinweise zur Problembehebung im Bereich FAQ der Job Scheduler Web Seiten.

Wahl des richtigen JDBC-Treibers

Der verwendete JDBC-Treiber muss mit der Version der Datenbank korrespondieren. Wenn der JDBC-Treiber nicht richtig gewählt wurde, hat der *Job Scheduler* Probleme mit der Verbindung zur Datenbank, die in folgender Fehlermeldung münden können:

Fehler SOCKET-61 ECONNREFUSED Connection refused (TCP-Port nicht erreichbar - Läuft der Server?) [connect].

Wir können aus rechtlichen Gründen keine Empfehlungen geben, können aber unsere Erfahrungen mitteilen:

- **MySQL**
Für MySQL Version 4.x haben wir gute Erfahrungen mit `mysql-connector-java-3.1.8-bin.jar` gemacht.
- **PostgreSQL**
wir verwenden den PostgreSQL 7 JDBC Treiber im Setup-Programm, da die Version 8 des Treibers aktuell das Einrichten von Prozeduren via JDBC nicht unterstützt. Den Treiber in der Version 8 setzen wir für den Betrieb des Job Schedulers ohne derzeit bekannte Probleme ein.
- **SQL Server**
Für SQL Server kann der JDBC-Treiber `sqljdbc.jar` sowohl mit Version 2000 wie mit Version 2005 verwendet werden, jedoch funktioniert ein älterer Treiber nicht unbedingt mit SQL Server 2005.
- **Oracle**
uns sind keine Probleme unserer Software beim Betrieb des Treibers `ojdbc14.jar` mit Oracle 9.2 und Oracle 10g bekannt.

ANSI-Modus in MySQL

Bei Verbindungen mit der MySQL-Datenbank wird versucht diese in den ANSI-Modus umzuschalten. Der ANSI-Modus ist zum Betrieb unserer Software erforderlich, da wir zur breiten Unterstützung von Datenbanksystemen auf ANSI-kompatibles SQL angewiesen sind.

Das automatische Einschalten des ANSI-Modus funktioniert bei älteren Versionen 4.0.x von MySQL nicht. Sie müssen für diese Versionen den ANSI-Mode im Datenbankserver selbst einstellen. Öffnen Sie hierfür die Datei `my.cnf` und fügen dort in der Sektion `[mysqld]` den Eintrag `SQL_MODE=ANSI_QUOTES` hinzu oder fügen Sie dem Startscript des Datenbankservers den Parameter `--ansi` hinzu. Starten Sie den Datenbankserver nach dieser Änderung neu.

Datenbankverbindung bei MySQL geht verloren

Sie erhalten den Fehler:

```
Fehler bei Verbindung mit [host]:[port]: SOS-JAVA-105 Java-Exception
java.sql.SQLException("No operations allowed after connection closed."),
methode=rollback []
```

Wenn die Verbindung zur MySQL-Datenbank eine längere Zeit offen ist, ohne dass der Job Scheduler auf sie zugreift, dann schliesst MySQL die Verbindung ohne dies dem Client (Job Scheduler) mitzuteilen. Sie können den Wert der MySQL-Systemvariablen `wait_timeout` erhöhen. Dieser Wert gibt die Zeit bis zum Schließen einer nicht interaktiven Verbindung in Sekunden an, wenn die Verbindung nicht genutzt wird.

Alternativ kann als Abhilfe ein regelmäßig laufender Job verwendet werden, z.B. der Job `scheduler_dequeue_mail`, der den Versand zwischengespeicherter eMails veranlasst, falls zuvor der Mail Server nicht verfügbar war. Dieser Job-Start veranlasst den Job Scheduler, einen History-Eintrag in die Datenbank zu schreiben, unabhängig davon, ob Mails zu versenden sind.

JDBC Connection bei SQL Server

Wenn Sie ältere JDBC-Treiber verwenden (z.B. `msbase.jar`, `mssqlserver.jar`, `msutil.jar`), dann wird die Verbindungszeichenfolge in der Datei `./config/factory.ini` anders formuliert als bei neuerem `sqljdbc.jar`

ALT:

```
db      =      jdbc      -class=com.microsoft.jdbc.sqlserver.SQLServerDriver
jdbc:microsoft:sqlserver://localhost:1433;selectMethod=Cursor;databaseName=scheduler
-user=scheduler -password=scheduler
```

NEU:

```
db      =      jdbc      -class=com.microsoft.sqlserver.jdbc.SQLServerDriver
jdbc:sqlserver://localhost:1433;sendStri
ngParametersAsUnicode=false;selectMethod=cursor;databaseName=scheduler
-user=scheduler -password=scheduler
```

Bitte beachten Sie die unterschiedlichen Klassennamen und die Groß-/Kleinschreibung beim Wert "cursor".

Email-Versand funktioniert nicht

Die Absenderadresse muss eine gültige Adresse für den Mail Server sein. Es kann andernfalls Probleme mit dem Hostnamen der Email-Adresse für ausgehende Mails geben. Passen Sie die Absenderadresse im Eintrag `log_mail_from` in der Datei `./config/factory.ini` an.

Wann ist ein (Neu)Start des Job Schedulers nötig?

Nach Änderungen einer Konfigurationsdatei aus dem `./config` Verzeichnis.

Nach dem Austausch von Bibliotheken im `./lib` Verzeichnis.

Im Routine-Betrieb ist ein Neustart des Job Schedulers - auch über einen längeren Zeitraum hinweg - nicht erforderlich.

Der Job Scheduler fragt nach einem Lizenzschlüssel, ist die Software nicht Open Source?

Wenn Sie den Job Scheduler unter Unix mit der Datei `./bin/scheduler` bzw. unter Windows mit der Datei `./bin/scheduler.exe` starten, erhalten Sie folgende Fehlermeldung:

```
SOS-1000 No licence key was found or licence key has expired. Please contact your
systems administrator or Software- und Organisations-Service GmbH, Fax +49 (30) 861
33 35, Mail info@sos-berlin.com [Scheduler].
```

Benutzen Sie zum Starten des Job Schedulers das Startskript `./bin/jobscheduler.sh` (Unix) bzw. `./bin/jobscheduler.cmd` (Windows). Die Binärdatei wird vom Startskript parametrisiert aufgerufen. Einer der Parameter (`-sos.ini=...`) im Startskript adressiert die Lizenzdatei `./config/sos.ini`, die einen freien Lizenzschlüssel für die GPL Version des *Job Scheduler* enthält.

Es gibt keinen funktionalen Unterschied zwischen GPL und kommerzieller eingesetzter Version, der Lizenzschlüssel hilft uns lediglich Kunden mit kommerziellem Support bei Rückfragen zu identifizieren.

Index

D

Datenbank einrichten 20, 5, 6, 14, 15, 20, 39

Datenbank entfernen 29, 30

Datenbank überwachen 6

Datenbankeinstellungen importieren 17, 17

Datenbankobjekte einrichten 18

Datenbanktreiber einrichten 15, 39, 39

Datenbankverbindung konfigurieren 17

Dienst deinstallieren 30

F

factory.ini 10, 11, 14, 15, 17, 31

J

Job Scheduler starten 17, 17, 20, 20, 31, 31, 40, 40

Job Scheduler stoppen 20, 21, 30, 30

Job-Konfiguration editieren 17, 17, 31, 31

Job-Konfiguration validieren 17, 31

Jobs konfigurieren 7, 9, 16, 17, 27, 31, 31

U

Uninstaller 18, 29

W

Web Server 4, 19, 29, 30, 33

Web-Oberfläche konfigurieren 9, 14, 19, 20, 20, 27, 31, 31