Job Java Job

Zweck

Der **Java Job** kompiliert beziehungsweise lädt eine konfigurierbare Java-Klasse zur Laufzeit und führt deren öffentliche Methode run() aus. Dadurch können kundenspezifische Erweiterungen innerhalb der Webdesk-Laufzeit ausgeführt werden. Im mitgelieferten Job-Descriptor ist der Job konfigurierbar, aber standardmäßig nicht aktiv.

Voraussetzungen und Sicherheit

- Die konfigurierte Klasse benötigt einen öffentlichen parameterlosen Konstruktor und eine öffentliche Methode run().
- Der Code läuft innerhalb der Webdesk-Anwendung und kann auf verfügbare Webdesk-Services und Daten zugreifen. Es darf daher nur geprüfter und vertrauenswürdiger Quellcode verwendet werden.
- Für die dynamische Kompilierung muss die Laufzeit in einer JDK-Installation mit verfügbarem Java-Compiler ausgeführt werden.
- Fehlt eine kompilierte Klasse und soll Quellcode kompiliert werden, muss die Option **Write JAVA source file** aktiviert sein.

Konfiguration

Filename (including package declaration)	Vollqualifizierter Klassenname einschließlich Paket, zum Beispiel at.workflow.webdesk.custom.jobs.TestSystemOut. Daraus werden Pfad und FileName abgeleitet.
JAVA source	Der vollständige Quellcode der kundenspezifischen Klasse.
Use a transaction	Legt fest, ob der kundenspezifische Job innerhalb einer Datenbanktransaktion ausgeführt wird. Die Einstellung wird bereits vor dem eigentlichen Lauf aus der Job-Konfiguration gelesen.
Write JAVA source file	Schreibt den Quellcode unter WEB-INF/work/runtime in die zur Paketstruktur passende Java-Datei.

Ausführung

1. Die Job-Konfiguration wird vor dem Start gelesen.
2. Wenn **Write JAVA source file** aktiviert ist, wird der Quellcode im Runtime-Verzeichnis gespeichert.
3. Der Klassenlader verwendet zuerst eine bereits verfügbare Klasse. Andernfalls lädt er eine vorhandene kompilierte Klasse aus dem Runtime- beziehungsweise Klassenverzeichnis oder kompiliert den hinterlegten Quellcode.
4. Die geladene Klasse wird instanziiert und ihre Methode run() ausgeführt.

Unterbrechen

Der Webdesk-Job ist unterbrechbar. Damit auch der kundenspezifische Code auf eine Unterbrechung reagieren kann, muss die Klasse zusätzlich eine öffentliche Methode interrupt() bereitstellen. Ohne diese Methode kann der kundenspezifische Code nicht zuverlässig unterbrochen werden.

Fehlerdiagnose

- Ist weder eine erreichbare kompilierte Klasse noch eine passende Quelldatei vorhanden, schlägt das Laden fehl.
- Bei Kompilierungsfehlern werden die Compilerfehler als Jobfehler weitergegeben.
- Prüfen Sie bei Fehlern den vollqualifizierten Klassennamen, die Paketstruktur, die Option **Write JAVA source file**, den verfügbaren JDK-Compiler und das Job-Log.

Felder

Name	Wert
Modul	Portal & Organisation (po)
Webdesk Actionname	Java Job
Artefakt-Typ	Job