Job Java Job

Purpose

The **Java Job** compiles or loads a configurable Java class at runtime and executes its public method run(). This allows customer-specific extensions to be executed within the Webdesk runtime. In the job descriptor that ships with the product, the job is configurable but not active by default.

Prerequisites and security

- The configured class requires a public no-argument constructor and a public method run().
- The code runs within the Webdesk application and can access available Webdesk services and data. Only reviewed and trustworthy source code may therefore be used.
- For dynamic compilation, the runtime must be executed in a JDK installation with an available Java compiler.
- If a compiled class is missing and source code is to be compiled, the option **Write JAVA source file** must be activated.

Configuration

Filename (including package declaration)	Fully qualified class name including package, for example at.workflow.webdesk.custom.jobs.TestSystemOut. The path and FileName are derived from it.
JAVA source	The complete source code of the customer-specific class.
Use a transaction	Determines whether the customer-specific job is executed within a database transaction. The setting is read from the job configuration before the actual run.
Write JAVA source file	Writes the source code to the Java file matching the package structure under WEB-INF/work/runtime.

Execution

1. The job configuration is read before the start.
2. If **Write JAVA source file** is activated, the source code is saved in the runtime directory.
3. The class loader first uses a class that is already available. Otherwise it loads an existing compiled class from the runtime or class directory, or compiles the stored source code.
4. The loaded class is instantiated and its method run() is executed.

Interrupting

The Webdesk job is interruptible. So that the customer-specific code can also react to an interruption, the class must additionally provide a public method interrupt(). Without this method, the customer-specific code cannot be reliably interrupted.

Troubleshooting

- If neither a reachable compiled class nor a matching source file exists, loading fails.
- In the case of compilation errors, the compiler errors are passed on as job errors.

- In the event of errors, check the fully qualified class name, the package structure, the option **Write JAVA source file**, the available JDK compiler and the job log.

Felder

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