

# Job JobSchedulerManagedDatabaseJob

- [JITL: The Managed Database Job](#)
- [MySQL \(MariaDB\)](#)
- [MS SQL Server](#)
- [Oracle](#)
- [PostgreSQL](#)
- [IBM/DB2](#)

## JITL: The Managed Database Job

### **i** Important

This job uses the deprecated `sos.scheduler.managed.JobSchedulerManagedDatabaseJob` that has been deprecated with Release 1.12. The [JobSchedulerManagedDatabaseJobSOSHibernate](#) job is provided as an alternative.

This job is used to execute (SQL-)statements in a database and can be used standalone or triggered by orders - i.e. as an order job. It can execute database procedures or SQL statements.

(A general overview of all JITL jobs can be found [Library of Standard Jobs - JITL](#)).

The description `JobSchedulerManagedDatabaseJob.xml` is stored in the `./jobs` directory of the JobScheduler installation.

The SQL instructions are written in the `command` parameter.

It is possible to define more than one instruction in the `command` parameter.

Such instructions are then carried out in the order in which they are written and must be separated by a semicolon and a subsequent new line.

You can use `&#10;` as newline.

The following is an example of a job chain in which the command contains two statements separated by `&#10;`.

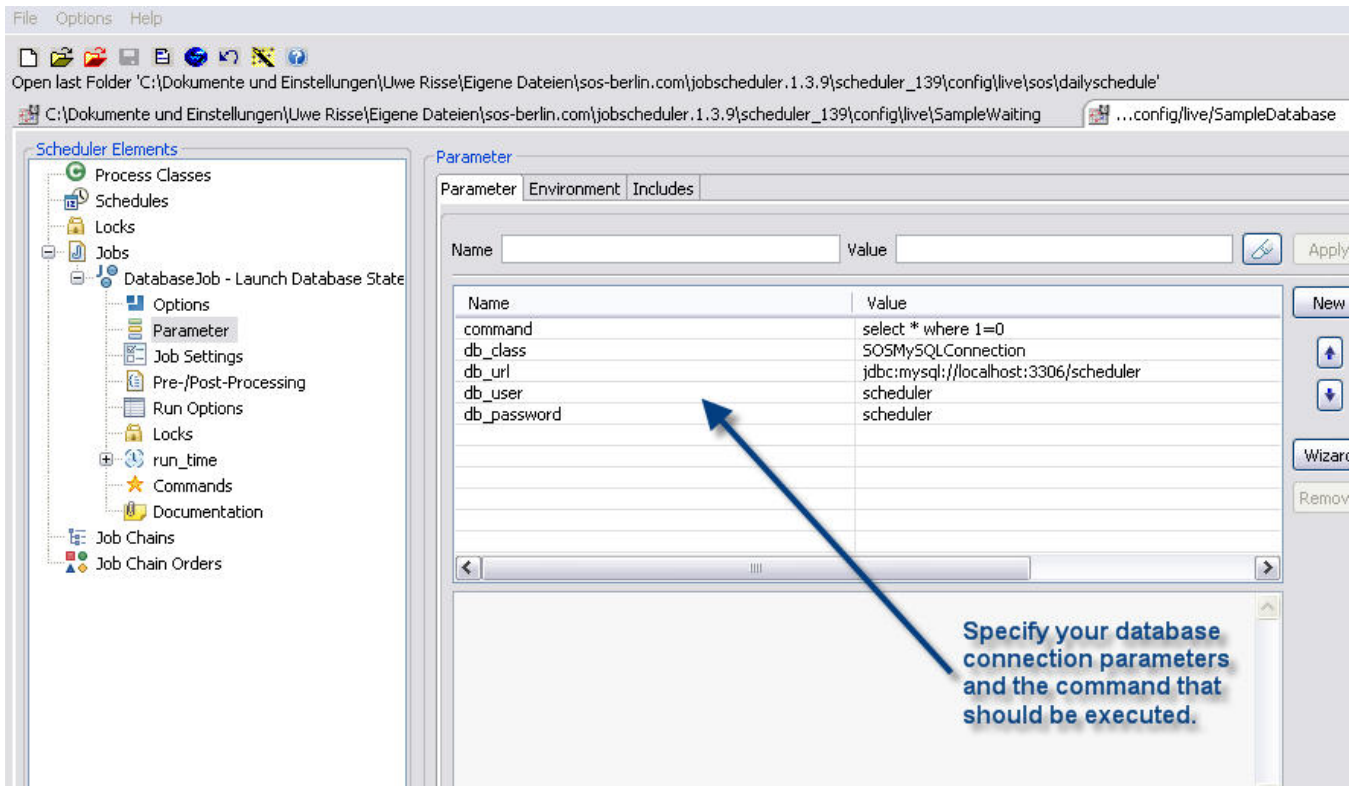
The screenshot shows the JobScheduler configuration interface. On the left, a tree view shows the hierarchy: Scheduler Elements > Jobs > Job: launchDB > Steps/Nodes. The main area is titled 'Chain Nodes for jobchain\_db' and contains a table with the following data:

State	Node	Job Dir	Next state	Err
launch	Job	launchDB	success	err
success	Endnode			
error	Endnode			

A callout box with the text 'A job chain containing the launchDB Job' points to the 'launch' row in the table. Below the table, there are sections for 'File Order Sources for jobchain\_db' with fields for Directory, Regex, and Max.

The job chain `launchDB.job_chain.xml`.





## MySQL (MariaDB)

```
<job title="Launch Database Statement" order="yes" stop_on_error="no">
  <description>
    <include file="jobs/JobSchedulerManagedDatabaseJob.xml"/>
  </description>
  <params>
    <param name="db_class" value="SOSMySQLConnection"/>
    <param name="db_driver" value="com.mysql.jdbc.Driver"/>
    <param name="db_url" value="jdbc:mysql://localhost:3306/scheduler"/>
    <param name="db_user" value="scheduler"/>
    <param name="db_password" value="scheduler"/>
    <param name="command" value=""/>
  </params>
  <script language="java" java_class="sos.scheduler.managed.JobSchedulerManagedDatabaseJob"/>
  <run_time/>
</job>
```

For other database management systems you have to modify the connection parameter like in the following examples:

## MS SQL Server

```

<job title="Launch Database Statement" order="yes" stop_on_error="no">
  <description>
    <include file="jobs/JobSchedulerManagedDatabaseJob.xml" />
  </description>
  <params>
    <param name="db_class"          value="SOSMSSQLConnection" />
    <param name="db_driver"        value="com.microsoft.sqlserver.jdbc.SQLServerDriver" />
    <param name="db_url"           value="jdbc:sqlserver://localhost:1433;
sendStringParametersAsUnicode=false;selectMethod=cursor;databaseName=scheduler" />
    <param name="db_user"         value="scheduler" />
    <param name="db_password"     value="scheduler" />
    <param name="command"         value=" " />
  </params>
  <script language="java" java_class="sos.scheduler.managed.JobSchedulerManagedDatabaseJob" />
  <run_time/>
</job>

```

## Oracle

```

<job title="Launch Database Statement" order="yes" stop_on_error="no">
  <description>
    <include file="jobs/JobSchedulerManagedDatabaseJob.xml" />
  </description>
  <params>
    <param name="db_class"          value="SOSOracleConnection" />
    <param name="db_driver"        value="oracle.jdbc.driver.OracleDriver" />
    <param name="db_url"           value="jdbc:oracle:thin:@localhost:1521:scheduler" />
    <param name="db_user"         value="scheduler" />
    <param name="db_password"     value="scheduler" />
    <param name="command"         value=" " />
  </params>
  <script language="java" java_class="sos.scheduler.managed.JobSchedulerManagedDatabaseJob" />
  <run_time>
</job>

```

See also:

- [How to run Oracle Stored Procedures using PL/SQL](#)
- [How to work with PL/SQL and the Oracle DBMS](#)
- [How to work with SQL\\*Plus and the Oracle DBMS](#)

## PostgreSQL

```

<job title="Launch Database Statement" order="yes" stop_on_error="no">
  <description>
    <include file="jobs/JobSchedulerManagedDatabaseJob.xml" />
  </description>
  <params>
    <param name="db_class"          value="SOSPgSQLConnection" />
    <param name="db_driver"        value="org.postgresql.Driver" />
    <param name="db_url"           value="jdbc:postgresql://localhost:5432/scheduler" />
    <param name="db_user"         value="scheduler" />
    <param name="db_password"     value="scheduler" />
    <param name="command"         value=" " />
  </params>
  <script language="java" java_class="sos.scheduler.managed.JobSchedulerManagedDatabaseJob" />
  <run_time>
</job>

```

## IBM/DB2

```
<job title="Launch Database Statement" order="yes" stop_on_error="no">
  <description>
    <include file="jobs/JobSchedulerManagedDatabaseJob.xml" />
  </description>
  <params>
    <param name="db_class"           value="SOSDB2Connection" />
    <param name="db_driver"         value="com.ibm.db2.jcc.DB2Driver" />
    <param name="db_url"            value="jdbc:db2://localhost:50000/scheduler:driverType=2;
retrieveMessagesFromServerOnGetMessage=true;" />
    <param name="db_user"           value="scheduler" />
    <param name="db_password"       value="scheduler" />
    <param name="command"           value="" />
  </params>
  <script language="java" java_class="sos.scheduler.managed.JobSchedulerManagedDatabaseJob" />
  <run_time>
</job>
```