Configuring the database connection

See also: Customizing the database connection

The database connection is handled by Hibernate. The properties below (set in **/properties/config.properties**) are used by the Hibernate configuration files **/properties/dao/hibernate/hibernate-jdbc.cfg.xml** and **/properties/dao/hibernate/hibernate-jndi.cfg.xml**, as well as the Tomcat configuration file **conf/server.xml** for quick-start deployments.

Creating the database

esup-helpdesk, when running the ant task init-data, creates all the structures (tables) of the database, but the database must exist. It is up to the deployer to create the database and make sure that the database user declared can create structures (tables, fields) and has read/write access to the database.

Use InnoDB with MySql

With MySql, the type of the database must be InnoDB.In order to automatically create InnoDB databases, set this property in the MySql configuration:

```
default-table-type=innodb
```

Database manager properties

Driver

The driver of the database manager (by default MySql JDBC):

```
#hibernate.connection.driver_class=com.mysql.jdbc.Driver
```

For Postgres use:

```
hibernate.connection.driver_class=org.postgresql.Driver
```

Hibernate dialect

The Hibernate dialect, by default MySql InnoDB:

```
#hibernate.dialect=org.hibernate.dialect.MySQLInnoDBDialect
```

For Postgres use:

```
hibernate.dialect=org.hibernate.dialect.PostgreSQLDialect
```

JDBC properties

esup-helpdesk always uses JDBC to access the database from ant tasks (asynchronous commands).

URL (required)

The JDBC URL of the database, by default **helpdesk** on **localhost**:

```
#hibernate.connection.jdbc.url=jdbc:mysql://localhost/helpdesk
```

For Postgres use:

```
hibernate.connection.jdbc.url=jdbc:postgresql://localhost:5432/helpdesk
```

Authentication (required)

The credentials used to connect to the database:

```
#hibernate.connection.jdbc.username=admin
#hibernate.connection.jdbc.password=secret
```

JNDI properties

esup-helpdesk can use JNDI to access the database from web requests. Using JNDI is recommended in production for performance reasons, it allows deployers to monitor the database load thanks to tools such as LambdaProbe (see 25 Surveillance des applications).

To use JDBC to access the database even from web requests, set this property:

```
hibernate.useJndi=false
```

To use a connection pool set like this on Tomcat in the application context,

```
<Resource
    name="jdbc/esup-helpdesk"
    auth="Container"
    type="javax.sql.DataSource"
    username="admin" password="secret"
    driverClassName="com.mysql.jdbc.Driver"
    url="jdbc:mysql://localhost/rennes1-20080905?autoReconnect=true"
    maxActive="100" maxIdle="10" maxWait="10000"
    poolPreparedStatements="true" validationQuery="SELECT 1"
    removeAbandoned="true" removeAbandonedTimeout="60"
    logAbandoned="true" />
```

Tell the application the name of the pool:

```
hibernate.useJndi=true
hibernate.connection.jndi.datasource=jdbc/esup-helpdesk
```

The application automatically detects its running mode (batch or web) to use JDBC (batch) or JNDI (web).

JDBC connector

esup-helpdesk is configured by default to use MySql, it embeds the MySql connector for Java (**mysql-connector-java.jar**). To connect to another database manager:

- use the JDBC driver and the Hibernate dialect that corresponds to your database manager,
- add the corresponding connector to the **/webapps/WEB-INF/lib** folder*,*
- allow the automatic recovering of this file when upgrading (see Recovering previous configuration and customizations when upgrading).

See: The Hibernate documentation

MySql

The MySql connector is embedded by default in the distribution.

Postgres

- Version 7.4: **pg74.1jdbc3.jar**
- Version 8.2: **postgresql-8.2-506.jdbc3.jar**

Connectors for Postgres can be downloaded from http://jdbc.postgresql.org/download.html.

Oracle

TODO Any Oracle deployer?