2c8 Server functionality

Table of contents

1 Introduction

2c8 Server functionality is a component that runs in an application server called Payara. Payara is an open source software derived from Oracle's Glassfish. Data is stored in MySQL or Microsoft SQL Server[1].

1. It is possible to use the express versions if necessary.

2 Database configuration

If this is a new installation of 2c8 Server functionality you will need to setup a database for it to run against. For existing installations the database will be upgraded when deploying the new server component.

2.1 Creating a new database

This chapter briefly describes how to set up a database for 2c8 Server functionality.

2.1.1 MySQL

Log on to mysql

```
# mysql -u root -p
```

Create a new database

```
mysql> CREATE DATABASE mt_server;
```

Add access to the database for user 'conciliate'

```
mysql> GRANT ALL PRIVILEGES ON mt_server.* TO
'conciliate'@'localhost' IDENTIFIED BY 'choose a password';
```

Force a reload of grant tables

```
mysql> FLUSH PRIVILEGES;
```

After logging off mysql, execute the script that populates the database

```
# mysql -u root -p mt_server < sql/mysql/mt_mysql_5.0.sql
```

2.2 Upgrading an existing database

The installation package contains patch scripts for upgrading existing databases to the current version. These will be applied automatically when the server component is upgraded in a later step.

3 Payara/GlassFish

2c8 Server functionality runs in an application server called Payara. Starting with 2c8 Modeling Tool 5.0 you will need version 5 of Payara. If you have an older version of Payara or are still running on the older Glassfish server you will need to uninstall this server and install Payara instead.

3.1 Prerequisites

To run Payara, JDK8 is required.

3.1.1 Debian/Ubuntu

Add the PPA repository, update and then install:

```
# sudo add-apt-repository ppa:webupd8team/java -y
```

```
# sudo apt-get update
```

```
# sudo apt-get install oracle-java8-installer oracle-java8-
set-default libmysql-java
```

3.2 Files needed

The following files are needed during the installation[2]:

Name	File
Payara	Installation files
Server component	MTServer-EAR-5.0-uX.ear

2. The installation procedure require root-access to the server.

3.3 Installation

3.3.1 Debian/Ubuntu

Download the payara-zip linked above to the server. The file you have downloaded should be named something like »payara-5.x.zip« (depending on the version, the actual filename might differ).

Unzip the file to an appropriate location (e.g /usr/share/):

```
# unzip payara-5.x.zip /usr/share/
```

3.4 JDBC Driver

Before configuring GlassFish you need to install the proper JDBC driver for your selected database.

3.4.1 MySQL Connector

Go to http://dev.mysql.com/downloads/connector/j/ and download the JDBC Driver, either as a zip or tar archive.

Unpack the downloaded content and copy the jar-file to the lib folder where Payara was installed. (/path/to/payara5/glassfish/lib/)

3.4.2 Restart Payara/Glassfish

After the jdbc-driver has been copied to the Payara/Glassfish installation a restart of the Payara service is required.

3.4.2.1 GNU/Linux

The Payara binaries are located at »/path/to/payara5/glassfish/bin/«. Change directory to that folder and run the asadmin tool to stop the service:

```
# sudo ./asadmin stop-domain domain1
```

When the service has stopped, start it again:

```
# sudo ./asadmin start-domain domain1
```

3.5 Database Connection

Open your server's admin console at http://{your server IP}:4848/ (alternatively http://localhost:4848 if you're accessing from within the server).

By default, there is no password set for the admin user to access the admin console. This is potentially a risk and we strongly encourage you to set a password.

Choose Resources » JDBC » Connection Pools from the tree on the left and click New… to create a new connection pool.

3.5.1 MySQL

Enter these values:

Name	mt-server
Resource type	javax.sql.DataSource
Database vendor	MySQL

Click Next and enter these values

General Settings	
Datasource Classname	com.mysql.cj.jdbc.MysqlDataSource
Transaction	
Transaction Isolation	read-committed
Additional Properties [3]	
user	conciliate[4]
password	conciliate[5]
databaseName	mt_server[6]
serverName	localhost[7]
portNumber	3306
useSSL	false

Click finish.

A new connection pool named mt-server has been created. Click on it to edit it further. Navigate to the »Advanced« tab and edit the following properties:

Connection Validation	Required
Validation method	table
Table name	version

3. Other attributes should be deleted. To do this, select all except the mentioned above and press the Delete Properties.

4. The database user

5. The database password

6. The name of your selected database

7. The name of the server where your database is located

Click Save.

3.5.2 Validate database connection

Choose Resources » JDBC » Connection Pools » mt-server from the tree. Click the Ping button. You should see a message saying Ping Succeeded if the application server connected successfully to the database.

3.5.3 Add JDBC Resource

Choose Resources » JDBC » JDBC Resources from the tree. Click New…to create a new JDBC Resource.

Enter these values

JNDI Name	jdbc/mt-server
Pool Name	mt-server

Make sure Enabled is checked, then click Ok.

3.6 Users/Security

3.6.1 Security Realm

Choose Configurations » server-config » Security » Realms from the tree. Click New…to create a new realm.

Enter these values

Name	conciliate
Class name [8]	com.sun.enterprise.security.auth.realm.jdbc.JDBCRealm
JAAS Context	jdbcRealm
JNDI	jdbc/mt-server
User table	users
User name column	user_id
Password column	password
Group table	realm_groups
Group name column	group_id
Assign group	user
Digest algorithm	MD5
Encoding	hex

8. Choose in list

Charset	UTF-8

Click Ok.

Choose Configuration » server-config » Security and choose:

Default Realm:	conciliate

Click Save.

3.7 Configure outgoing mailserver in 2c8 Server functionality (Optional)

This configuration step allows the 2c8 Server functionality to send emails to users and subscribers, which is a feature used by 2c8 Workflow and the notification system when publishing. If you plan on using one of these features to send automatic mail notifications an outgoing mail server must be configured in 2c8 Server functionality. Note that this configuration is optional and only necessary if you want mail notifications.

Choose Resources – JavaMail Sessions and click on New....

Create a new JavaMail Session with the JNDI name mail/workflow. Enter the mandatory values:

Mail Host	Hostname for the outgoing mail server
Default User	User account for login to the mail server
Default Sender Address	Mail address to send mail from

The basic properties to use are:

mail.smtp.auth	true (Use authentication for mail server login)
mail.smtp.password	12345 (Password for authentication)
mail.smtp.starttls.enable	true (Use STARTTLS to handle encrypted communication channel)
mail.smtp.port	587 (Mailserver port to use)
mail.always.same.sender	true or false (Set to "true" if mail should appear to be sent from workflow@2c8.com, set to "false" if mail should appear to be sent from Default Sender Address)

Please note that the values to use differ between different mail servers. The next sections describe some examples of settings to use for some common email services.

3.7.1 Microsoft Online

Demands that ”Default Sender Address” is the same as ”Default User”.

Mail Host	smtp.office365.com
mail.smtp.auth	true
mail.smtp.password	Password for the account Default User
mail.smtp.starttls.enable	true
mail.smtp.port	587
mail.always.same.sender	true

3.7.2 Gmail

Demands that the Gmail account is accessible from “Less secure apps”.

Mail Host	smtp.gmail.com
mail.smtp.auth	true
mail.smtp.password	Password for the account Default User
mail.smtp.starttls.enable	true
mail.smtp.port	587
mail.always.same.sender	true or false

3.8 Deploy server component

Choose Applications from the tree. Click Deploy to install a new component (if you are upgrading, start by undeploying the previous component).

Click Packaged file to be uploaded to the server. Then click Browse... and choose the file »MTServer-EAR-5.1-uX.ear«.

Click OK to deploy the component. This will also upgrade the database to the current version so it could take a while for big databases. When it is done, restart the application server.