Installation guide for Windows and MySQL

2c8 Server functionality

conciliate

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 Microsoft SQL Server

If there is no available database you can use Microsoft SQL Server Express. It is a scaled-down version of SQL Server, available for free. There are some restrictions such as a maximum database size of 10GB and only one gigabyte of RAM usage.

2.1.1.1 Install Microsoft SQL Server Express (if needed)

- Start the installation program for **Microsoft SQL Server Express**.
- Choose **Mixed Mode** and enter a password for the system administrator (sa)
- Install **SQL Server Management Studio Express**.

2.1.1.2 Create database

- Start **SQL Server Management Studio Express** from the start menu.
- Choose **Connect** in the first dialog shown to connect to the database server.
- Right-click **Databases** in the tree on the left and choose **New Database**.
- On the **General** page, enter a database name, such as »mt_server«.
- On the **Options** page choose the appropriate collation.
- Click **OK** to create the database.
- From the menu choose **File » Open » File…**
- Open the database script »sql/mssql/mt_mssql_5.0.sql«

- Make sure that the database you created is selected as the active (through the drop-down menu).
- Click the **Execute** button.

2.1.1.3 Assign database user

- In **SQL Server Management Studio Express** expand the **Security** node in the tree and right-click on **Login**. Choose **New Login…** from the popup menu.
- On the **General** page, enter a **Login name**. In this example we use »conciliate«.
- Choose **SQL Server authentication** and enter a password. In this example we use »conciliate«.
- Make sure that **Enforce password policy** and **Enforce password expiration** are **not** selected so that the password does not expire.
- Choose the created database as **Default database**.
- On the **User Mapping** page, choose **db_owner** for the created user in this database.
- Click **OK** to create the login.

2.1.1.4 Enable TCP/IP in SQL Server

- Start **SQL Server Configuration Manager**. It is found in the start menu under **SQL Server » Configuration Tools » SQL Server Configuration Manager**.
- Choose **Protocols for SQLEXPRESS**.
- Right-click **TCP/IP** and choose **Enable** from the popup menu.
- Right-click **TCP/IP** again and choose **Properties** from the popup menu.
- Under **IPAll** choose **TCP Port** and enter »1433«.
- Click **OK** to close the dialog.
- Choose **SQL Server Services** in the tree.
- Right-click **SQL Server (SQLEXPRESS)** and choose **Restart** from the popup menu.

2.1.1.5 Create a database in Microsoft SQL Server

If SQL Server Express was installed in previous steps, these steps should be omitted.

- To allow the application server to connect to the database **SQL Server Authentication** needs to be activated. In this example we use port number »1433« to connect to the database. You can verify that the connection is working by using telnet from the computer running the application server.
- Create a login to the database. In this example we have used »user=conciliate« and »password=conciliate«.

- Create a database. In this example it is named »mt_server«.
- Set the created login as **db owner** on the database.
- Run the script »sql/mssql/mt_mssql_5.0.sql« on the empty database.

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 Microsoft Windows

We recommend Redhat's OpenJDK which can be downloaded from RedHats's download page.

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

3.3 Installation

3.3.1 Microsoft Windows

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 C:\Program Files\).

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

3.3.1.1 Setup service

If you want Payara to run as a Windows service, open a command prompt as Administrator and navigate to the payara-bin folder (e.g /path/to/payara5/bin/). Execute the following command:

```
asadmin.bat create-service
```

A service named »domain1 Payara Server« is created.

3.4 JDBC Driver

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

3.4.1 Microsoft SQL Server

- Download Microsoft JDBC Driver 6.0 for SQL Server
- Run the self-extracting zip sqljdbc_6.0.7728.100_enu.exe and unpack somewhere on your harddrive.
- Copy the file sqljdbc42.jar 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 Windows

To restart the Payara/Glassfish service, use the installed windows service. If you did not install a windows service, use the shortcuts located in the Start-Menu.

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 Microsoft SQL Server

Enter these values:

Name	mt-server
Resource type	javax.sql.DataSource
Database vendor	Microsoft SQL Server

Click **Next** and enter these values

General Settings	
Datasource Classname	com.microsoft.sqlserver.jdbc.SQLServerDataSource
Transaction	
Transaction Isolation	read-committed
Additional Properties [3]	
user	conciliate[4]
password	conciliate[5]
databaseName	mt_server[6]
serverName	localhost[7]
portNumber	1433[8]

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
Table name	version

Click Save.

---

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

8. 1433 is default when not using cluster

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 [9]	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
Charset	UTF-8

9. Choose in list

Click Ok.

Choose **Configuration » server-config » Security** and choose:

Default Realm: conciliate

Click **Save.**

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

For 2c8 Workflow to be able 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-5.0-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.