Installation and Configuration - v22

Planforge Server, Version 22

This document describes the whole installation or upgrade process for a "Planforge" server, including all current configuration options.

Contents

1. System Requirements

1.1. System Requirements for the Server

For setting up a Planforge server you will need the following components:

- A supported operating system
- The Java SE Runtime Environment
- The application-server "Tomcat"
- A compatible database

The following tables show in detail which versions of the components above you can use:

Operating System	- Microsoft Windows Server 2012, 2016, 2019 - Linux 2.4 or later - Apple Mac OS X Snow Leopard (v10.6) or later
System Memory (RAM)	Depending on which combination of database and operating system was used, these are the minimum recommendations [*]:
Hard Drive Space	Approximately 1 GByte (Initial size for the Tomcat application server and the deployed web-application)

	PostgreSQL	Oracle	Microsoft SQL Server
Windows 7/8/Server or later	3 GBytes	4 GBytes	4 GBytes
Linux [**]	2 GBytes	3 GBytes	N/A
Mac OS X	3 GBytes	4 GBytes	N/A

[*] Measured using a clean installation and both, the application-server and the database were installed on the same machine which is highly recommended.
[**] Pure server-distribution without graphical user-interface. For Linux distributions with KDE/Gnome and X-server an additional Gigabyte RAM might be needed

Java Platform	OpenJDK (recommended):
	- Latest: 16 Update 1 or later - Also supported: 15 Update 2 or later, 14 Update 1 or later, 11 Update 5 or later, 8 Update 111 or later
	Oracle Java:
	- Latest: 16 Update 1 or later - Also supported: 15 Update 2 or later, 11 Update 2 or later, 10 Update 1 or later, 8 Update 25 or later
	⊘ Incompatible Java Platforms - Java SE 5 (1.5.0), Java SE 6 (1.6.0), Java SE 7 (1.7.0) and Java SE 9 (1.9.0) - OpenJDK 7 Update 95 (1.7.0_95) or later updates of release 7, JRE or JDK
Application Server	Apache Tomcat installed using the original distributions from https://tomcat.apache.org/
	- Latest: 9.0.28 or later - Also supported: 8.5.48 or later
	⊘ Incompatible Tomcat Versions - Tomcat 10 or later - Tomcat 8.0 or earlier - Linux: Preinstalled distributions or installed by package-managers
Database	- PostgreSQL (Unicode/UTF-8 character set) - Recommended database - Latest: 13.2 - Also supported: 12.7 or later, 11.1 or later, 10.4 or later, 9.0.7 or later - Oracle (Unicode/UTF-8 character set) - Latest: 19c - Also supported: 18c, 12c - Microsoft SQL Server - Latest: SQL Server 2016 - Also supported: SQL Server 2014, 2012
	⊘ Unsupported or Incompatible Databases "Express"-editions of Oracle and Microsoft SQL Server - Earlier versions of the databases listed above

1.2. System Requirements for Clients

Generally, a supported operating system including a compatible web-browser is required to access a Planforge server:

Operating System	
	• Microsoft Windows 7, 8, 8.1, 10 • Linux 2.4 or later • Apple Mac OS X Snow Leopard (v10.6) or later
System Memory	1 Gigabyte RAM

Compatible Web-Browsers:

- Windows: Edge, Google Chrome, Mozilla Firefox, Internet Explorer 11 (IE 11 is highly recommended when using Internet Explorer)
- Linux: Google Chrome, Mozilla Firefox
- Mac OS X: Safari, Google Chrome, Mozilla Firefox

Browser Configuration:

- Enable JavaScript if disabled
- Let your browser accept "Session-Cookies" (Including "Third Party Cookies")
- Add an exception for popup-blockers (Including ad-blockers or similar) to not enable them for your server's URL (required for downloading attachments, reports and following links)
- Configure your browser to print background images of webpages (Only required for printing Gantt-charts or other graphical views)
    - Mac OS X (all browsers): "Print background images" and "Print background colors" is usually a default UI element of the "Print"-menu
    - Windows:
        - Firefox: Firefox menu - "Print" - Page Setup - Activate "Print Background (colors & images)"
        - Internet Explorer: Gear-button (upper right corner) - "Print"-menu - "Page Setup" - Activate "Print Background Colors and Images"
        - Google Chrome: Usually no explicit setting is required, although using the latest release is recommended as this is one of the recently added functionalities

2. How to Install Java

If you are not sure which version is installed, please execute the following command in your terminal or command line:

```
java -version
```

This should show something like "Java(TM) SE Runtime Environment" along with a version/build-number. If this number does not match those listed among our system requirements in the previous chapter ("Java Platform"), then please install a newer version.

If the command does not work at all, then most likely no Java was installed yet. If this is the case, please use the links below to setup Java.

For large production-servers please always use the JDK-packages, for smaller installations or trial-setups the JRE-package will be sufficient. Depending on which operating system you are using, these steps should be followed to install or update Java:

Windows	The latest Java-releases can be downloaded from:
	- OpenJDK: https://jdk.java.net/ - Oracle JDK: http://www.oracle.com/technetwork/java/javase/downloads/index.html ⚠ **Note for Tomcat memory settings** Make sure to install the 64-bit variant of Java, as it is highly recommended to configure at least 2048MB for your Tomcat memory settings.
Linux	The latest Java-releases can be downloaded from: - OpenJDK: https://jdk.java.net/ - Oracle JDK: http://www.oracle.com/technetwork/java/javase/downloads/index.html Alternatively, use the package manager of your Linux-distribution to install a new version of OpenJDK or Oracle Java. Instructions on how to install Java and the Java Plug-in can be found on the download page.
MacOS X	- Mac OS X releases earlier than OS X "Lion" (10.7.x) already include Java. To upgrade simply run a software update - Users of Mac OS X "Lion" (10.7.x) and later: After the latest software-updates for these operating systems you will need to download Java from: - OpenJDK: https://jdk.java.net/ - Oracle JDK: http://www.oracle.com/technetwork/java/javase/downloads/index.html

3. How to Install Tomcat

After Java was installed, the "Tomcat" application-server is required to host our Planforge web application - This section describes how to install and configure a basic "Tomcat"-server.

> ⚠️ **Installation of Planforge in its own Tomcat**
>
> Planforge should be installed in its own Tomcat instance. If you are using an existing Tomcat instance, please install a second one for Planforge.

3.1. Tomcat on Windows

First download the latest supported Tomcat **"Core Binary Distribution"** from http://tomcat.apache.org/. The supported versions are listed in the "System Requirements for Server"-section.

To install, make sure you are logged in as a user with administrator privileges and simply double click the installer (If the installation fails, then you might need to explicitly start the installer by right-clicking it and choosing "Run as Administrator"). The installer will guide you through the whole installation-process.

Afterwards, Tomcat has to be configured for our needs. Please open the "Configure Tomcat" program (If you cannot find it, execute "Monitor Tomcat" from the new Program Group which was created during install and afterwards double click the new icon in your taskbar)

1. Select the "Java"-tab inside the "Configure Tomcat"-tool.
2. For "Initial memory pool" or "Minimum memory pool" enter the value "2048"
3. For "Maximum memory pool" enter the value "2048" - Please note that higher memory settings are highly recommended for productive installations, see chapter 3.4 Tomcat Configuration
4. If you are using Java 16 or later, make sure the following lines are included in the "Java 9 Options" field (should be included by default in Tomcat 9.0.49 and later):

```
--add-opens=java.base/java.lang=ALL-UNNAMED
--add-opens=java.base/java.io=ALL-UNNAMED
--add-opens=java.base/java.util=ALL-UNNAMED
--add-opens=java.base/java.util.concurrent=ALL-UNNAMED
--add-opens=java.rmi/sun.rmi.transport=ALL-UNNAMED
```

5. Save the changes by confirming the dialog-window with the "Apply"- or "OK"-button

To start Tomcat, please again open the "Configure Tomcat" dialog - The buttons which control Tomcat can be found in the "General"-tab. After Tomcat was started, you can open a web-browser and try to access Tomcat's welcome-page (If needed, replace "localhost" with your server's IP or hostname and "8080" with the port you are using):

```
http://localhost:8080/
```

If that page cannot be loaded, please check the "stdout"-logfiles in Tomcat's "logs"-folder. If these don't show an error, then most likely a firewall or similar is blocking the access.

⚠️

⚠️ **Planforge home directory and user for Tomcat**

Planforge uses the home-folder of the user-account that starts Tomcat's service to store the configuration and other related files, therefore Tomcat should always be started with the same user. Users of Windows operating systems will usually start Tomcat as a "System"-service which will create this folder in "C:\Windows\system32\config\systemprofile\planforge\".

You can change the user that starts Tomcat, in: "Control Panel" > "Administrative Tools" > "Services" - Locate the "Apache Tomcat"-service in the list of services and double click it. Afterwards you can select a user-account in the tab "Log on". Alternatively, you can set this path manually by following these steps:

1. Open the "Environment Variables" and create a new **system variable**.
2. Set the name to ONEPOINT_PROJECT_HOME and enter the desired path as the value of the variable.
3. Restart Tomcat.
4. Again connect to the web-application using your web-browser and check, if a folder was created in the desired location.

⚠️ **File Encoding for Tomcat**

To avoid any possible inconsistencies with the file encoding we would like to recommend to use UTF-8 as file encoding. This can be added to the "setenv.bat"-file which is available in Tomcat's "bin"-folder or in the "Java"-tab of the "Configure Tomcat"-tool.

3.2. Tomcat on Linux

Today's Linux-distributions usually come with a preinstalled Tomcat-server or generally, your favourite package-manager will allow you to install one automatically. However, our web-application will only deploy and work correctly if Tomcat was installed manually from scratch, using the original distributions - This guide should work for all current Linux-distributions:

1. Create a "tomcat" user-account, being member of a group "tomcat" using a terminal/shell (If your distribution doesn't have "groupadd" or "useradd"-commands, try "adduser" or "addgroup" instead):

```
sudo groupadd tomcat
sudo useradd -g tomcat -s /usr/sbin/nologin -m tomcat
```

⚠️ If your Linux-distribution doesn't have the file "nologin" in "/usr/sbin", then it can most likely be found in "/sbin" instead (You can use the commands "which" or "whereis" to retrieve the full path to the file).

The "-m"-switch will automatically create a home-directory for the new user-account (Will for example be created as "/home/tomcat" if your users' home-folders are stored in "/home"). We will need it for storing our configuration and related files.

2. Next we can proceed to download the latest "Core Binary Distribution" of Tomcat from http://tomcat.apache.org/ - This guide will use the file names of Tomcat 8, but there won't be any differences if you are using later releases.

3. Extract the files using the command-line:

```
tar xvf apache-tomcat-8.0.xx.tar.gz
```

4. Paste the following lines into a text-editor and save them with the filename "setenv.sh" - Please note that higher memory settings are highly recommended for productive installations, see chapter 3.4 Tomcat Configuration.

```
#!/bin/sh
#

export JAVA_OPTS='-Djava.awt.headless=true -Xms2048m -Xmx2048m'
```

5. If you are using Java 16 or later and a Tomcat version older than 9.0.43, make sure to also add following line to the "setenv.sh"-file:

```
export JAVA_OPTS='--add-opens=java.base/java.lang=ALL-UNNAMED --add-opens=java.base/java.io=ALL-UNNAMED
--add-opens=java.rmi/sun.rmi.transport=ALL-UNNAMED --add-opens=java.base/java.util=ALL-UNNAMED --add-
opens=java.base/java.util.concurrent=ALL-UNNAMED'
```

6. In the extracted folder, there'll be a directory called "bin". Move the file "setenv.sh" to that location:

```
mv setenv.sh apache-tomcat-8.0.xx/bin/
```

7. Alter the permissions of the folder so our "tomcat" user-account and members of the group "tomcat" are allowed to access it:

```
sudo chown -R tomcat:tomcat apache-tomcat-8.0.xx
```

8. Alter the name to something simpler and move the folder to the location you want to install it (We'll use the location "/usr/local" and the name "tomcat")

```
sudo mv apache-tomcat-8.0.xx /usr/local/tomcat
```

Now you can try to start Tomcat - Switch to your new "tomcat"-user:

```
sudo su - -s /bin/sh tomcat
```

As user "tomcat":

```
cd /usr/local/tomcat/bin
./catalina.sh start
```

If everything works fine you should see something like this:

```
...
Apr 16, 2012 6:48:23 AM org.apache.coyote.AbstractProtocol start
INFO: Starting ProtocolHandler ["http-bio-8080"]
Apr 16, 2012 6:48:23 AM org.apache.coyote.AbstractProtocol start
INFO: Starting ProtocolHandler ["ajp-bio-8009"]
Apr 16, 2012 6:48:23 AM org.apache.catalina.startup.Catalina start
INFO: Server startup in 668 ms
```

⚠

> ⚠️ **Error "java.net.BindException: Address already in use"**
>
> If an error "java.net.BindException: Address already in use" shows up, check if the ports "8080" or "8005" are already in use. If this is the case, you can alter these port-numbers in the file "usr/local /tomcat/conf/server.xml". Afterwards try to start Tomcat again.

Now you can open a web-browser and try to access Tomcat's welcome-page (Replace "localhost" with your server's IP or hostname and "8080" with the port you are using):

```
http://localhost:8080/
```

Afterwards you should see Tomcat's welcome page, otherwise check if your firewall or similar (For example "iptables" or "SELinux") is blocking the port.
To stop the server, press the keys <CTRL> and <C> together. And if you need to switch back to your regular user-account:

```
exit
```

To start and stop Tomcat later on, you could directly execute the scripts/commands "startup.sh" and "shutdown.sh" in Tomcat's "bin"-folder.

> 🛑 **Never start Tomcat with "root"-permissions!**
>
> Please make sure to never start Tomcat with "root"-permissions (Or via "sudo") - The application-server must always be started or stopped using a dedicated user-account (for example with the user "tomcat"), otherwise the file-permissions inside Tomcat's directory-structure will get mixed up.

> ⚠️ **Planforge home directory and user for Tomcat**
>
> Additionally, Planforge uses the home-folder of the user-account that starts Tomcat to store the configuration and other related files. This path can also be set manually by following these steps:
>
> 1. Edit the file "setenv.sh" in the "bin"-directory of Tomcat and add the line "export ONEPOINT _PROJECT_HOME [PATH]"
> 2. Replace [PATH] with the directory, where the configuration etc. should be stored
> 3. Restart Tomcat
> 4. Again connect to the web-application using your web-browser and check, if a folder was created in the desired location.

3.3. Tomcat on Mac OS X

If Tomcat is not preinstalled on your operating system, please download the latest supported Tomcat **"Core Binary Distribution"** from http://tomcat.apache.org/. The supported versions are listed in the "System Requirements for Server"-section. Afterwards follow these steps:

1. First let's find out which group- and user-IDs below 500 are free (IDs above 500 are reserved to normal users). Open a terminal/shell and execute the following two commands:

```
dscl . -list /Groups PrimaryGroupID
dscl . -list /Users  UniqueID
```

2. Choose a number below 500 which can neither be found in the results for the first command nor in the results for the second one. I'll use "444" - So please make sure to replace "444" with your own number in the following commands:

```
sudo dscl . -create /Groups/tomcat PrimaryGroupID    444
sudo dscl . -create /Groups/tomcat RealName          "Tomcat Group"
sudo dscl . -create /Groups/tomcat Password          "*"
sudo dscl . -create /Users/tomcat  UniqueID          444
sudo dscl . -create /Users/tomcat  PrimaryGroupID    444
sudo dscl . -create /Users/tomcat  NFSHomeDirectory  /Users/tomcat
sudo dscl . -create /Users/tomcat  UserShell         /usr/bin/false
sudo dscl . -create /Users/tomcat  RealName          "Tomcat User"
sudo dscl . -create /Users/tomcat  Password          "*"
```

3. If you want to verify that everything was stored correctly, you can query the user account with the following command:

```
dscl . -read /Users/tomcat
```

> ⚠ **"tomcat"-account shown at Mac OS X Lion Login Form**
>
> If the "tomcat" user-account shows up at the login-form, then execute the following command in the terminal:
>
> ```
> sudo dscl . -delete /Users/tomcat/ AuthenticationAuthority
> ```
>
> Afterwards the user-account will be hidden from the login-page.

4. Next create the home-folder for our user-account and grant permissions to the Tomcat-user and group:

```
sudo mkdir /Users/tomcat
sudo chown tomcat:tomcat /Users/tomcat
```

5. Next we can proceed to download the latest "Core Binary Distribution" of Tomcat from http://tomcat.apache.org/ - This guide will use the file names of Tomcat 8, but there won't be any differences if you are using later releases.

6. Extract the files using the command-line

```
tar xvf apache-tomcat-8.0.xx.tar.gz
```

7. Paste the following lines into a text-editor and save them with the filename "setenv.sh" - Please note that higher memory settings are highly recommended for productive installations, see chapter 3.4 Tomcat Configuration.

```
#!/bin/sh
#

export JAVA_OPTS='-Djava.awt.headless=true -Xms2048m -Xmx2048m'
```

8. If you are using Java 16 or later and a Tomcat version older than 9.0.43, make sure to also add following line to the "setenv.sh"-file:

```
export JAVA_OPTS='--add-opens=java.base/java.lang=ALL-UNNAMED --add-opens=java.base/java.io=ALL-UNNAMED
--add-opens=java.rmi/sun.rmi.transport=ALL-UNNAMED --add-opens=java.base/java.util=ALL-UNNAMED --add-
opens=java.base/java.util.concurrent=ALL-UNNAMED'
```

9. In the extracted folder, there'll be a directory called "bin". Move the file "setenv.sh" to that location:

```
mv setenv.sh apache-tomcat-8.0.xx/bin/
```

10. Alter the permissions of the folder so our "tomcat" user-account and members of the group "tomcat" are allowed to access it:

```
sudo chown -R tomcat:tomcat apache-tomcat-8.0.xx
```

11. Alter the name to something simpler and move the folder to the location you want to install it (We'll use the folder "/Library" and the name "Tomcat")

```
sudo mv apache-tomcat-8.0.xx /Library/Tomcat
```

Now you can try to start Tomcat - Switch to your new "tomcat"-user:

```
sudo -s -u tomcat
```

As user "tomcat" enter:

```
cd /Library/Tomcat/bin
./catalina.sh start
```

If everything works fine you should see something like this:

```
05.04.2012 20:30:06 org.apache.coyote.AbstractProtocol start
INFO: Starting ProtocolHandler ["http-bio-8080"]
05.04.2012 20:30:06 org.apache.coyote.AbstractProtocol start
INFO: Starting ProtocolHandler ["ajp-bio-8009"]
05.04.2012 20:30:06 org.apache.catalina.startup.Catalina start
INFO: Server startup in 1108 ms
```

⚠️

> ⚠️ **Error "java.net.BindException: Address already in use"**
>
> If an error "java.net.BindException: Address already in use" shows up, check if the ports "8080" or "8005" are already in use. If this is the case, you can alter these port-numbers in the file "usr/local /tomcat/conf/server.xml". Afterwards try to start Tomcat again.

Now you can open a web-browser and try to access Tomcat's welcome-page (Replace "localhost" with your server's IP or hostname and "8080" with the port you are using):

```
http://localhost:8080/
```

Afterwards you should see Tomcat's welcome page, otherwise check if your firewall is blocking either the port or Java itself.

To stop the server, press the keys <CTRL> and <C> together. And if you need to switch back to your regular user-account:

```
exit
```

To start and stop Tomcat later on, you could directly execute the scripts/commands "startup.sh" and "shutdown.sh" in Tomcat's "bin"-folder.

> 🛑 **Never start Tomcat with "root"-permissions!**
>
> Please make sure to never start Tomcat with "root"-permissions (Or via "sudo") - The application-server must always be started or stopped using a dedicated user-account (for example with the user "tomcat"), otherwise the file-permissions inside Tomcat's directory-structure will get mixed up.

> ⚠️ **ONEPOINT Projects home directory and user for Tomcat**
>
> Additionally, Planforge uses the home-folder of the user-account that starts Tomcat to store the configuration and other related files. This path can also be set manually by following these steps:
>
> 1. Edit the file "setenv.sh" in the "bin"-directory of Tomcat and add the line "export ONEPOINT _PROJECT_HOME [PATH]"
> 2. Replace [PATH] with the directory, where the configuration etc. should be stored
> 3. Restart Tomcat
> 4. Again connect to the web-application using your web-browser and check, if a folder was created in the desired location.

3.4. Tomcat Configuration

If you have followed our guides in the previous chapters to setup Tomcat, then usually no further configuration will be required. The following chapter summarizes all Java Options that were applied:

Java Options/Java Heap Space

Parameter	Configured value (Mbytes)
"Initial/Minimum memory pool"/-Xms	2048m
"Maximum memory pool"/-Xmx	2048m

> ⊖ **Important Note for Memory Settings**
>
> These values are our default recommendations for all Planforge test environments and will initially reserve around 2GB main memory. For productive installations the following values can be used for the memory configuration of your Tomcat-Installation depending on the available resources.
>
> | **"Initial/Minimum memory pool"/-Xms** | 4096m | 5120m | 6144m | 8192m | 16384m | 24576m |
> |---|---|---|---|---|---|---|
> | **"Maximum memory pool"/-Xmx** | 4096m | 5120m | 6144m | 8192m | 16384m | 24576m |

> ⚠ **Additional Java Options for Linux/Mac OS X**
>
> For these operating systems it's recommended to always add the setting "-Djava.awt. headless=true"

The full overview on these parameters can be found in the Java SE documentation at http://www.oracle. com/technetwork/java/javase/tech/vmoptions-jsp-140102.html

If you didn't follow our installation guides in this document please compare the configuration above with those of your Tomcat-server and make sure the values are either identical or higher.

Depending on which operating-system and which kind of Tomcat-distribution was used the parameters above will usually always be read from these files or locations:

OS	"Core" Tomcat Binary Distribution	Configuration will be read from
Windows	"32-bit/64-bit Windows Service Installer"	Application "Configure Tomcat", tab "Java"
Windows	"32-bit" or "64-bit Windows zip"	File "setenv.bat", created in Tomcat's "bin"-folder
Linux	"zip" or "tar.gz"	File "setenv.sh", created in Tomcat's "bin"-folder
Mac OS X	"zip" or "tar.gz"	File "setenv.sh", created in Tomcat's "bin"-folder

The full documentation to all different distributions can be found at http://tomcat.apache.org/

SSL for Tomcat
To enable SSL for Tomcat, please follow the guides at:
http://tomcat.apache.org/tomcat-8.5-doc/ssl-howto.html (For Tomcat 8.5)
http://tomcat.apache.org/tomcat-9.0-doc/ssl-howto.html (For Tomcat 9)

3.5. Setting up web-sockets for Planforge

In Planforge some of the communication between the server and the client, especially keeping clients in sync with each other, is done via web-sockets, therefore it is essential to also set this up on your respective web server for the optimal user experience. The sections below will describe how to set up these web-sockets with an Apache web server specifically, but this can also be used as a reference in case you are using another web server like nginx.

Steps for setting up web-sockets with Apache

1. Add a symbolic link from ".../mods-available/proxy_wstunnel.load" to "mods-enabled".
2. Add a symbolic link from ".../mods-available/headers.load" to "mods-enabled".
3. Add the following snippet to your virtual-host or proxy .conf-file:

Apache .conf for passing web-socket-calls

```
<IfModule proxy_wstunnel_module>
        ProxyPass /planforge/ws/ ws://localhost:8080/planforge/ws/
</IfModule>
```

NOTE: Please be aware that you may need to adjust the following parameters in the example above to fit your environment:

planforge - If your "planforge.war" file has been renamed in Tomcat's "webapps"-directory
localhost - In case Apache and Tomcat are not running on the same system
:8080 - In case your Tomcat instance does not use the default port

If your configuration uses its own Content Security Policy (CSP), you might want to add the following entry to your virtual-host configuration:

Content Security Policy

```
<IfModule headers_module>
    ### ATTN: replace wss with ws if your are NOT using https (NOT recommended!)
    Header set Content-Security-Policy "connect-src 'self' wss://PLANFORGE_URL; default-src 'self'; style-src
'self' 'unsafe-inline'; script-src 'self' 'unsafe-inline' 'unsafe-eval'; img-src 'self
' data:; frame-src 'self' https://www.youtube.com"
</IfModule>
```

NOTE: Please replace "PLANFORGE_URL" above with the public URL, that is used to access your Planforge instance via a browser, e.g. wss://your-company.com/planforge. Additionally, only the part "connect-src 'self' wss://PLANFORGE_URL;" is required to enable the web-socket-pass-through, other entries in the example above show a working configuration for use with Planforge.

4. Database Setup

The Planforge web application stores all its data in a separate database. This chapter will show you how to prepare your favorite database server so we can write our data to it.

> ⚠ **Separate database for each Planforge instance**
>
> Please note that for each Planforge instance a separate database is required. If you are using a productive system as well as a test system, please create a separate database for each Planforge instance.

If you haven't chosen a database yet, then this is the moment to check the full list of supported databases in the system-requirements table at the beginning of this document and install a compatible database-server before continuing this guide.

If you only need to verify the version-number of an existing database-server, then please continue reading in the following sections, according to your database-type:

4.1. PostgreSQL

> ⚠ **Verifying the PostgreSQL version-number**
>
> To find out which PostgreSQL-version is already installed, you can simply sign on to your database-server using the "psql" command-line utility. If it afterwards only shows one version-number, then the client and server will have the same one. Otherwise it will show an explicit "Server" version-number.

PostgreSQL itself is available as a guided installer-package for all operating-systems - It can either be obtained directly from http://www.postgresql.org/ or it can be installed using a package-manager. During installation you will most likely be asked to select a character-set - If "UTF-8" is not among the available choices, please make sure to keep the "Default"-setting.

PostgreSQL Default Timezone:
After the installation was completed, set the default timezone of the database to "GMT":

1. Shutdown your PostgreSQL database instance
2. Locate the file **postgresql.conf** in the "data"-directory of your PostgreSQL database
3. Open the file using any text editor (e.g. notepad, gedit or textedit) and locate the following line:

   ```
   #timezone = unknown          # actually, defaults to TZ
   ```

4. Change this line in the following way:

   ```
   timezone = GMT               # actually, defaults to TZ
   ```

5. Afterwards start your PostgreSQL database instance
   If your PostgreSQL server fails to start after the change, please make sure that the permissions on the file "postgresql.conf" are correct

(Continued on next page)

Creating the user-account and the database:
Finally, to be able to connect to PostgreSQL we will need an empty database and a user-account which is allowed to access it. These are the basic steps for the PostgreSQL console, assuming your database's default user is called "postgres":

1. Open the PostgreSQL "SQL Shell" or open a command-line/terminal window and type

```
psql -U postgres
```

2. Enter the password which you specified when installing PostgreSQL and proceed with the following commands to create a database called "planforge", belonging to the user "planforge" and with password "planforge"

```
postgres=# create user planforge password 'planforge';

postgres=# create database planforge owner planforge encoding 'utf-8';

postgres=# \q
```

4.2. Microsoft SQL Server

> ⚠ **Verify MS SQL Server's version-number**
>
> The most comfortable way to verify MS SQL's version-number is to open the "SQL Server Management Studio" and to sign on to your database. The version-number will then be shown in brackets besides the database-icon or in the "Server Properties"-dialog.

To be able to connect to MS SQL we need a "login", an empty database and a database user to connect to it.

Please make sure that you are logged into the database with permissions to create the required elements, for example as a local administrator user (Windows Authentication) or as the user "sa" (SQL Authentication).

The required queries (Use "SQL Server Management Studio" or the "sqlcmd" command-line to enter them):

1. First we'll have to create an empty database "planforge":

```
CREATE DATABASE planforge;
```

2. For the best compatibility with Planforge, we highly recommend to configure your database to use a case sensitive collation (further information can be found here), therefore we will now check which collation the created database is using by running the following command:

```
SELECT name, collation_name FROM sys.databases WHERE name = 'planforge';
```

If the collation has "..._CS_..." in the name, it is already case sensitive and does not have to be adjusted. However if it includes "..._CI_...", please check which collation is the case sensitive equivalent of your current one and set it for your database. For example, if your collation is "Latin1_General_CI_AS", you can set it to the recommended collation with the following command:

```
ALTER DATABASE planforge COLLATE Latin1_General_CS_AS;
```

3. Next we need a login:

```
CREATE LOGIN planforge WITH PASSWORD = 'planforge_secret', DEFAULT_DATABASE = planforge;
```

4. And the following command connects to our new database and creates a database-user "pf_user" which will be linked to the login "planforge"

```
USE planforge;
CREATE USER pf_user FOR LOGIN planforge;
```

5. And finally we'll need to grant the role "db_owner" to the user "pf_user":

```
EXEC sp_addrolemember 'db_owner', 'pf_user';
```

⚠ **Switching from the jTDS to Microsofts JDBC driver**

For new instances, Microsofts JDBC driver is used by default, however instances that were set up with release 19.2.1.1 or earlier may still use the old jTDS driver, which can lead to issues with newer releases of ONEPOINT Projects (before release 22) or Planforge (release 22 and later). We recommend updating the configuration to use the new driver by following these steps:

1. Stop your Tomcat-instance and navigate to the ONEPOINT Projects/Planforge Home directory, which contains the configuration "cofiguration.oxc.xml" and more.
2. Back your current configuration up and edit the configuration file afterwards.
3. Replace the line "<database-driver>net.sourceforge.jtds.jdbc.Driver</database-driver>" with "<database-driver>com.microsoft.sqlserver.jdbc.SQLServerDriver</database-driver>"
4. Replace the line "<database-url>jdbc:jtds:sqlserver://SQLSERVER:PORT/onepoint OR planforge</database-url>" with "<database-url>jdbc:sqlserver://SQLSERVER:PORT; database=onepoint OR planforge</database-url>"
   Please note here, that the format of additional parameters is different between the drivers, for example "database=planforge" instead of "/planforge". Details for all parameters can be found here.
5. Save the changes, start your Tomcat-instance and try connecting to ONEPOINT Projects /Planforge to test the configuration.

4.3. Oracle

> ⚠️ **Verifying Oracle's version-number**
>
> If you're not sure which Oracle-release you are currently using you can check the version number by simply signing on to your database using the "SQL*Plus" command-line utility. Afterwards Oracle will show the full version-number of the database you are currently connected to.

There are two general requirements to be able to connect to an Oracle database-server:

1. The correct "JDBC Thin" database-driver according to your database- and Java-version must be copied to Tomcat's "/lib" folder (This driver-file can always be found in Oracle's download-area)
2. A user-account must be created in Oracle, allowing to sign on to your database, with the following roles (Only "Standard"-roles are needed, not "Admin"):
   - CONNECT
   - RESOURCE

These are the exact permissions which will be granted by these roles:

Role	Permissions
CONNECT	CREATE SESSION
RESOURCE	CREATE CLUSTER, CREATE INDEXTYPE, CREATE OPERATOR, CREATE PROCEDURE, CREATE SEQUENCE, CREATE TABLE, CREATE TRIGGER and CREATE TYPE

Here is an example on how to create a database user from Oracle's SQL*Plus command-line - (We will not specify a tablespace and use the default one).

1. Make sure you are logged in as sysdba and execute the following:

```
create user planforge identified by planforge_secret;
```

   Will create a database user 'planforge' with the password 'planforge_secret'.
2. Next we will need to grant permissions to that user-account:

```
grant connect, resource to planforge;
```

   Will grant connect and resource roles to the user "planforge".

> ⚠️ **"ORA-28000: the account is locked" error**
>
> If you receive an "ORA-28000: the account is locked" error-message, you will have to unlock the account like in the following example and repeat the 'grant' statement above:
>
> ```
> alter user planforge account unlock;
> ```

⚠️

⚠️ **"ORA-01000: maximum open cursors exceeded error"**

If this error occurs, modify the program to use fewer cursors. If this error occurs often, shut down Oracle, increase the value of OPEN_CURSORS, and then restart Oracle.

Checking currently open cursors (Must be run with dba permissions, e.g. "sys as sysdba"):

```
select max(a.value) as highest_open_cur, p.value as max_open_cur

from v$sesstat a, v$statname b, v$parameter p

where a.statistic# = b.statistic#

and b.name = 'opened cursors current'

and p.name= 'open_cursors'

group by p.value;
```

Increasing Open Cursors (Example for increasing to 5000):

```
ALTER SYSTEM SET open_cursors = 5000 SCOPE=BOTH;
```

5. Planforge - Installation or Upgrade

5.1. Planforge - Installation

There are currently two possibilities to configure all required settings to start your Planforge installation.

5.1.1. Configuration Wizard

Before starting the installation, please make sure that Tomcat is not running, then:

1. Copy the file "planforge.war" inside the downloaded ZIP-archive to Tomcat's "webapps"-folder
2. Start Tomcat
3. Connect to the new web-application using your browser (For example, if your Tomcat-server was running locally on port 8080, the URL would be "http://localhost:8080/planforge/" - Otherwise please replace 8080 with your Tomcat's port and use the correct IP-address or host name instead of "localhost")

When connecting to Planforge for the first time, the Configuration Wizard will be shown which is the easiest way to configure all required parameters for starting your Planforge installation. The Configuration Wizard consists of the following three steps:

Step 1 "Setup Language and License"

In the first step you are able to select the system language of Planforge and upload a valid license file.

Step 2 "Setup Database"

The second step allows you to configure the parameters for connecting to your database. The following databases are available for the field "Database Type":

- PostgreSQL
- Oracle
- Microsoft SQL Server

Setup Database

Database Type	PostgreSQL ▾
	The type of your database you want to connect to
Hostname	localhost
	Hostname or IP address of your database server.
Port	5432
	The TCP Port Number of your database server.
Database	planforge
	The name of your database to connect to.
Username	planforge
	The username used to access the database.
Password	planforge
	The password used to access the database.

Previous | Next

Further details on supported databases can be found in chapter "1.1 System Requirements for the Server"

Step 3 "Setup Demodata and Mail Server"

In the last step you can configure the following settings:

- The password for the administrator user can be set.
- By enabling the checkbox "Include Demodata" it is possible to start your installation with a preconfigured dataset.
- The SMTP Mail Server can be set up by activating the corresponding checkbox. If a mail server is configured Planforge is able to send notifications. The notifications that should be sent automatically can be customized in Planforge's system settings after the setup is finished.

5.1.2. Manual Installation

Preparing the Folder "planforge" for configuration files and more

Before proceeding please make sure that Tomcat was stopped, then:

1. Copy the file "planforge.war" inside the downloaded ZIP-archive to Tomcat's "webapps"-folder
2. Start Tomcat
3. Connect to the new web-application using your browser. For example, if your Tomcat-server was running locally on port 8080, the URL would be "http://localhost:8080/planforge" - Otherwise please replace 8080 with your Tomcat's port and use the correct IP-address or host name instead of "localhost"
4. Directly afterwards open Tomcat's main-logfile in a text-editor (Will either be called "catalina.out" (Linux) or "stdout..." (Windows) - It's the largest file in Tomcat's "logs"-folder)
5. Search for the string "user.home". This part of the log will contain a few environment settings along with the path like:

```
Environment Settings:
    java.vendor: Oracle Corporation
    java.version: 1.8.0_60
    ...
    user.home: /Users/tomcat/planforge/planforge/
    ...
```

   This line shows the path on the disk-volume where our web-application will store configuration-parameters, your license-file and similar. The folder-structure "planforge/planforge" will already have been created in this location if Tomcat has permission to write to the parent-directory shown in the line above. If the directory "planforge" and the subfolder "planforge" were not created, please make sure the user-account that starts Tomcat has write-permissions for the parent-directory, restart Tomcat and repeat step number "3." above.
6. Next stop Tomcat

Afterwards copy the license-file ("license.oxl.xml") which you received either attached to a delivery e-mail or downloaded from a link inside that message to the subfolder "planforge" inside "planforge". (Continued on next page)

Preparing the Configuration File

Inside the downloaded Planforge zip-archive you will find a folder "dbconfigs", containing three subfolders "postgresql", "mssql" and "oracle". Each one of these contains a configuration file called "configuration. oxc.xml".

Please pick the one according to the database-type you are using and copy it to the subfolder "planforge" inside "planforge" (Where you copied the license-file to).

If you were using default values during database installation and when preparing the database like described in the chapter "Database Setup", then the configuration-file should work right out of the box for your server.

Otherwise open the file "configuration.oxc.xml" with a text- or xml-editor and alter the configuration-details like described below (The example will show a PostgreSQL configuration, but the configurations for other database-types will look quite identical). Please keep all other lines not mentioned by these steps unchanged:

- This line stores the JDBC connection string inbetween `<database-url>` and `</database-url>` which describes the hostname/IP-address, the port and the database-name (Or Oracle's SID):

  ```
  <database-url>jdbc:postgresql://localhost:5432/planforge</database-url>
  ```

  You can find examples describing how this JDBC connection string works in the last chapter "APPENDIX" of this manual. Please alter it so it matches your database-server's connection properties.
- This line stores the login name of the database user that's allowed to connect to your database:

  ```
  <database-login>planforge</database-login>
  ```

  Please replace "planforge" with the correct login-name.
- This line stores the password of the database user that's allowed to connect to your database:

  ```
  <database-password encrypted="false">planforge</database-password>
  ```

  Please replace "planforge" with the correct one (The password will later automatically get encrypted during the first start of the web application)

Afterwards store the changes and proceed with the final step of the installation.

(Continued on next page)

Finishing the installation

1. Start Tomcat
2. Again connect to the web-application using your browser like you did during the first steps of this chapter

Now you'll have to wait a moment during the first startup. After approximately a minute the login will appear and you should be allowed to sign on as user "administrator" (Keep the password field empty).

If the login does not appear then please open Tomcat's main-logfile in a text-editor. It will contain explicit error-messages if something went wrong during startup. If the logfile doesn't show any issues please compare the "System Requirements for Clients" at the beginning of the document with your web-browser's configuration.

If you need help please contact our support-team by sending a mail to support@planforge.io (Please make sure to attach your Tomcat's logfile if this first startup fails and to include a small description what the web-browser shows when trying to sign on).

Loading the Demonstration Dataset

If you now want to load the demonstration-dataset that can be found within the folder "demodata" inside the downloaded Planforge zip-Archive (File "demodata.opbx"), you can do the following:

1. Copy the file "demodata.opbx" to the folder "planforge/planforge/backup"
2. Sign on as "administrator" at the Planforge login of your server
3. Afterwards open the system-settings toolset by clicking the tool-shaped button in the upper-right corner
4. Click the tool "Repository"
   The button "Restore" will afterwards allow you to select the file "demodata.opbx" and to restore it. Please note that this process will overwrite all existing data.

Restoring Existing Backups created in Earlier Releases

If you want to restore an existing backup file (Extension "opbx") created in earlier releases of Planforge, you can do that the same way like described above for loading the demonstration dataset. Simply copy the backup-file(s) from your existing server's folder "planforge/planforge/backup" to the directory "planforge/planforge/backup".

Afterwards the tool "Repository" will allow you to restore the files in that folder, but keep in mind that this process will overwrite all existing data you would have entered in the new release.

5.2. Planforge - Upgrade

Generally, the upgrade process is very simple and will allow you to upgrade your current instance to the most recent version without having to install any other version in between. However, there are still some rare cases where this is required, therefore please check which version of Planforge you are currently using by logging in with any user and opening the "About Planforge" dialog by clicking on your username in the top right. This dialog will show the exact software version including the software's build date. If your current version is listed in any of the sections below, please follow the corresponding section for any additionally required steps, otherwise you can install the latest version of Planforge by following these steps:

1. Stop your Tomcat-instance
2. Create a backup of your current Planforge database

> ⊘ **Database Backup**
>
> Please make sure to create a full backup of the database before proceeding with the upgrade-process. The fastest and safest way to do this will be to use your database-server's toolset:
>
> - PostgreSQL: Tool "pg_dump"
> - Oracle: Tools "exp" or "expdp"
> - MS SQL Server: Application "SQL Server Management Studio"

3. Delete or move the existing file `"planforge.war"` and the deployed folder `"planforge"` inside the directory `"webapps"` of the Tomcat-installation
4. Copy the new file `"planforge.war"` from the software distribution into the directory `"webapps"` of your Tomcat-server
5. Delete all files in Tomcat's `"temp"`-folder
6. Start the Tomcat-instance again and connect to Planforge as usual. After connecting, the "Starting up Planforge"-page will be shown while the upgrade process is running
7. After the upgrade process has finished, you can check the installed version by logging in with any user and opening the "About Planforge"-dialog by clicking on your username in the top right

> ⚠ **Upgrading to a new major release**
>
> When upgrading to a new major release (for example version 18.x to 19.x), please note that a new license file is also required, which customers will receive together with each major release (e.g. release 19.0). Please make sure to place the new license file in the "planforge/planforge"-directory in the home-directory of the user, that starts the Tomcat-instance, before starting the upgrade.

5.2.1. Upgrading from ONEPOINT Projects 18.0.1.1 or earlier to Release 19 or later

1. Stop the Tomcat application server
2. Create a backup of the database

> ⊘ **Database Backup**
>
> Please make sure to create a full backup of the database before proceeding with the upgrade-process. The fastest and safest way to do this will be to use your database-server's toolset:
>
> - PostgreSQL: Tool "pg_dump"
> - Oracle: Tools "exp" or "expdp"
> - MS SQL Server: Application "SQL Server Management Studio"

3. Delete the existing file `"onepoint.war"` and the deployed folder `"onepoint"` inside the directory `"webapps"` of the Tomcat-installation
4. Delete all files in Tomcat's `"temp"`-folder
5. Download a version of release 18.0.2 or 18.0.3
6. Copy the new file `"onepoint.war"` into the directory `"webapps"` of your Tomcat-server
7. Start Tomcat and connect with your web-browser as usual
8. Now it can take a few minutes until the login appears, as at this point, the database schema will be updated
9. Afterwards stop the Tomcat application server
10. Repeat steps 2-8, but with the "`onepoint.war`" (before release 22) or "`planforge.war`" (release 22 and later) file of release 19 or later instead.

If the login does not appear then please open Tomcat's main-logfile in a text-editor (This logfile will either be called "catalina.out" or "stdout..." - It's the largest file in Tomcat's "logs"-folder). It will contain explicit error-messages if something went wrong during startup. If the logfile doesn't show any errors please compare the "System Requirements for Clients" at the beginning of the document with your web-browser's configuration.

If you need help please contact our support-team by sending a mail to support@planforge.io (Please make sure to attach your Tomcat's logfile if this first startup fails and to include a small description what the web-browser shows when trying to sign on).

6. Planforge - Configuration

When starting the software for the first time, all database connection parameters are stored in a file called `"configuration.oxc.xml"`. It can be found in your `"planforge/planforge"` folder. The "planforge" folder was previously also named `"Onepoint Project Home"` or "onepoint" before release 22. This chapter sums up how to configure or add other configuration options in this file.

If you make changes to this file which are not allowed or will not work, then Tomcat's logfiles ("stdout" for Windows or "catalina.out" for Linux/Unix) and the log file of Planforge in "planforge/planforge/logs" will give a hint on what went wrong.

This is an example of what the configuration file will initially look like:

```
<?xml version="1.0" encoding="UTF-8"?>
<configuration>
     <database name="Default">
       <database-type>PostgreSQL</database-type>
       <database-driver>org.postgresql.Driver</database-driver>
       <database-url>jdbc:postgresql://localhost:5432/planforge</database-url>
       <database-login>planforge</database-login>
       <database-password encrypted="true">dGNlam9ycHBvZXJlbmlttZWVhcnA=</database-password>
     </database>
</configuration>
```

Everything between `<database name="Default">` and `</database>` describes your working database connection and should remain unchanged.

To add other configuration options to this file which are not automatically written to it, please open it with a texteditor or xml editor and insert the additional settings between the closing `</database>` and `</configuration>` entries. For example:

```
     </database>
    <smtp-server>192.168.1.10</smtp-server>
    <max-attachment-size>40</max-attachment-size>
</configuration>
```

6.1. Enabling the Notification System

```
<smtp-from>Planforge</smtp-from>
<smtp-from-email>planforge@localhost</smtp-from-email>
<smtp-prefix>[PLANFORGE]</smtp-prefix>
<smtp-server>localhost</smtp-server>
<smtp-port>25</smtp-port>
<smtp-tls>true</smtp-tls>
<smtp-username>test-user</smtp-username>
<smtp-password encrypted="false">pass</smtp-password>
```

If the SMTP-Server is running on a different system, please replace "localhost" with the hostname or IP of your SMTP Server and save the file. After restarting Tomcat, the notification system will be enabled (Of course, you will need to enter e-mail addresses for all your user accounts in the "Users" tool - and afterwards enable the needed notifications in the "Notifications"-tool inside the Planforge user interface)

> ⚠️ **Using STARTTLS instead of TLS**
>
> If TLS does not work or is not sufficient for the authentication with your mail server, you can also try using STARTTLS instead with the following entry:
>
> ```
> <smtp-starttls>true</smtp-starttls>
> ```
>
> Please make sure to only use either TLS or STARTTLS, but not both at the same time.

6.2. Notification Trigger

Some notifications like for example "Scheduled work was not yet started" are sent at the same time every day at 8:00 AM GMT.

This can be overridden by the following setting (Here an example to send the notifications at 12'clock, every day):

```
<notification-trigger>0 0 12 * * ?</notification-trigger>
```

The full documentation on the the format of the "Cron-Expression" string inbetween <notification-trigger> and </notification-trigger> can be found here:

http://www.quartz-scheduler.org/docs/tutorials/crontrigger.html

6.3. Starting Planforge automatically

This setting allows to start Planforge automatically after Tomcat has been started. To enable this function, two entries are required:

```
<enable-auto-start>true</enable-auto-start>
<connect-url>http://localhost:8080/planforge</connect-url>
```

The "`connect-url`" should contain the URL, at which Planforge will be available within your network or from the internet, as this entry is also used when communicating with other applications. For example, for our cloud servers, this entry would look like this: <connect-url>https://europe.planforge.io/</connect-url>

6.4. Session Timeout Configuration

This setting allows to configure the session-timeout for all signed on users in the file "configuration.oxc. xml":

```
<http-session-timeout>60</http-session-timeout>
```

The session-timeout is specified in minutes, the above example would extend the session-timeout to 60 minutes (Default is always 30 minutes).

6.5. CMIS Session Timeout

```
<cmis-session-timeout>720</cmis-session-timeout>
```

This configuration-property is specified in minutes (Default is 12 hours / 720 minutes) and controls the timeout for CMIS-connections which can be configured in the tool "ADMINISTRATE/External Apps", tab "CMIS".

6.6. Backup Folder Location

```
<backup-path>PATH</backup-path>
```

Please replace `"PATH"` with the full path to your backup folder - For example

```
<backup-path>D:\planforge\backups</backup-path>
```

6.7. Altering the name of the log-file

If required, the name of the logfile in the folder `"planforge/planforge/planforge.log"` can be altered by adding the following configuration-parameter:

```
<log-file>NEW_FILE_NAME.TXT</log-file>
```

Please replace "NEW_FILE_NAME.TXT" with a filename and extension of your choice.

6.8. LDAP Authentication

Enabling an LDAP authentication will be more complex in comparison to other configuration properties described in this chapter, so let's start with the main structure of the configuration inside the file `"config uration.oxc.xml"`.

> ⚠️ **Example Files**
>
> Following this guide will be easier with an example configuration. You will find three example configuration-files inside the folder `"ldapconfigs"` of the downloaded Planforge zip-archive. One for Active Directory, one for other LDAP implementations like OpenLDAP, ApacheDS or similar, and the third one in the "extended" sub-directory contains basically a copy of the Active Directory configuration and contains examples to show additional possible configuration options.

All LDAP configuration-properties will be entered inbetween the tags `<ldap>` and `</ldap>`, consisting of four main-sections:

```
  </database>

  <ldap>
    <connection>
                <!-- Contains the LDAP-server's connection parameters -->
    </connection>
    <update-schedule>
      <!-- Configures how often the LDAP-synchronization
                  cycle will be initialized -->
    </update-schedule>
    <users>
      <!-- Configures which user-accounts will be retrieved
            from the directory server and how their attributes
            will be mapped to Planforge -->
    </users>
    <groups>
      <!-- OPTIONAL - Configures which groups will be retrieved
            from the directory server and how their attributes
            will be mapped to Planforge -->
    </groups>
  </ldap>

</configuration>
```

The sections `<connection>`, `<update-schedule>` and `<users>` are mandatory. The section `<group s>` will only have to be added if you need LDAP-groups to be retrieved from your directory server. If you don't need to synchronize LDAP-groups, you can drop the whole `<group> ... </group>` part.

The following describes each of the separate sections of the LDAP-configuration in detail. The example configurations apply to Microsoft's Active Directory unless a different LDAP implementation is mentioned.

6.8.1. LDAP Connection Parameters ("<connection>")

```
<connection>
    <connect-url>ldap://ad.example.com:389</connect-url>
    <!-- <keystore>path/to/java/keystore</keystore> -->
    <!-- <dns-url>dns://somehost/wiz.com</dns-url> -->
    <security-authentication>simple</security-authentication>
    <security-principal>cn=Administrator,cn=Users,dc=ad,dc=example,dc=com</security-principal>
    <security-credentials encrypted="false">secret</security-credentials>
    <security-protocol>plain</security-protocol>
    <!-- <password-hash-algorithm>SHA</password-hash-algorithm>-->
</connection>
```

All connection properties in detail:

<connect-url>	The URL to connect to the LDAP server. The URL has to start with either "ldap" or "ldaps". Default port for ldap is 389, for ldaps 636.
<keystore>	The path to the Java keystore. See keytool for further information.
<dns-url>	The DNS host and domain names to use.
<security-authentication>	The authentication mechanism to use. Possible values are `"none"`, `"simple"`, *sasl_mech*, where *sasl_mech* is a space-separated list of SASL mechanism names. See SSL for further description of these strings.
<security-principal>	Specifies the name of the user doing the authentication.
<security-credentials encrypted="false">	Specifies the credentials of the user doing the authentication. The entered information will be encrypted during the next start of Planforge and the entry will be set to "<security-credentials encrypted="true">"
<security-protocol>	This property is only necessary if SSL authentication is required. To activate SSL-authentication set the value to "ssl", otherwise set it to "plain".
<password-hash-algorithm>	The password hashing algorithm to use. Do not specify this propertiy if the password hashing algorithm is stored within LDAP's "userPassword" field. Normally you would need to set this property.
<connection-timeout>	This timeout cancels the connection to LDAP if the system is not accessible. The timeout values are in seconds and if no value is set, the default of 10 seconds is used.

6.8.2. Update Scheduler settings ("<update-schedule>")

These settings are used to control the periodical updates. These updates will shadow any changes (if configured) from the LDAP server to the Planforge database.

Typical updates are: every day at 5 AM from Monday to Friday and this is exacly what the following example configuration does:

```
<update-schedule syncNewUsers="true">
  <minute>0</minute>
  <hour>5</hour>
  <month>*</month>
  <day-of-week>1-5</day-of-week>
 <!-- or: <day-of-month>1,10,20</day-of-month> -->
</update-schedule>
```

⚠ **Synchronizing new created users from LDAP to Planforge**

For a successful synchronization of new created users from LDAP to Planforge, the attribute "syncNewUsers" has been implemented. This attribute is optional and possible values for this attribute are "true" and "false" (default = "false"):

1. If set to "false" or not specified, then users newly created in LDAP are not automatically shadowed in Planforge. These users will be shadowed if the administrator opens the "Users" -tool in the System Settings and presses the "Synchronize"-button there. If a non-shadowed user logs in to Planforge, then this user is automatically shadowed. Setting the syncNewUsers attribute to "false" makes the administrator more aware when new users are imported and also prevents Planforge from automatically using licenses for new LDAP users.
2. If set to "true", then all users matching the <search-filter> (see below) will be automatically shadowed in Planforge. Setting this attribute to "true" allows resource administrators to link resources to new users without having to ask the administrator to import these new users.

The allowed values in detail:

Key	Allowed Values	Allowed Special Characters
<minute>	0-59	, - * /
<hour>	0-23	, - * /
<month>	1-12 or JAN-DEC	, - * /
<day-of-week>	1-7 or SUN-SAT	, - * / L #
<day-of-month>	1-31	, - * / L W

Either day-of-week or day-of-month must be specified. The values/special-characters that can be entered are actually separate fields of a "cron Expression". You can find the full description to these fields at: http://www.quartz-scheduler.org/docs/tutorials/crontrigger.html

6.8.3. User Configuration ("<users>")

Before you continue with this section please note that Planforge is licensed on a per-user basis and each user-account synchronized from LDAP represents one licensed user. To check how many user-accounts can still be created inside your Planforge server:

1. Sign on as "administrator" at the login-form of your Planforge server
2. Open the system settings by clicking the tool-shaped button in the upper-right corner
3. Click the tool "License" in the tool-dock on the left side

The lines "Number of Observer Users", "Number of Contributor Users" and "Number of Manager Users" will show in brackets the maximum count of user-accounts for each user-level. The number on the left will show how many user-accounts have already been created for each user-level.

To make sure that only these user-accounts get synchronized/retrieved that are really required, please make sure to make these user-accounts which shall have access to your Planforge server members of an LDAP-group which we can specify later in this configuration-section.

The section describing the user-retrieval and mapping is initiated with the `<users>` tag and closes with `</users>` - Here's an example what this section could look like:

```
<users>
  <!--    <signon-pattern>DOMAIN\:username</signon-pattern> -->
  <uuid>objectGUID</uuid>
  <retrieval>
    <search-filter>(objectclass=user)</search-filter>
    <search-base>cn=Users,dc=ad,dc=example,dc=com</search-base>
    <search-scope>subtree</search-scope>
  </retrieval>
  <mapping>
    <OpUser.Name value="sAMAccountName"/>
    <OpUser.Password value="userPassword"/>
    <OpUser.FirstName value="givenName"/>
    <OpUser.LastName value="sn"/>
    <OpUser.Email value="mail"/>
    <OpUser.Description value="description"/>
    <OpUser.Phone fixed="false" synched="true" value="telephoneNumber"/>
    <OpUser.Mobile fixed="false" synched="true" value="mobile"/>
    <OpUser.Fax fixed="false" synched="true" value="facsimileTelephoneNumber"/>
    <OpUser.GroupMembership value="memberOf"/>
    <!-- Optional automatically create resources linked to users
    <OpUser.LinkedResource synched="false">
        <OpResource.Name value="sAMAccountName"/>
        <OpResource.FullName value="displayName"/>
        <OpResource.Description value="description"/>
    </OpUser.LinkedResource>        -->
    <!-- Optional Language Mapping
      <OpUser.Language fixed="true" value="en" synched="false" /> -->
  </mapping>
</users>
```

"<signon-pattern>" and "<uuid>"

<signon-pattern>	The user sign on pattern may be used for e.g. signing on to a domain. The value `:username` will be replaced by the signing-on username.
<uuid>	This field is required to track user name changes. Not all LDAP implementations offer a unique identifier for all entries inside the directory server, but if your LDAP server offers such an attribute then please make sure to specify it like shown in this example (In the example above Active Directory's "objectGUID"-attribute was used)

The "<retrieval>" sub-section

This sub-section inside `<users>` defines which user-accounts shall be retrieved, an example:

```
<retrieval>
  <search-filter>(objectClass=user)</search-filter>
  <search-base>cn=Users,dc=ad,dc=example,dc=com</search-base>
  <search-scope>subtree</search-scope>
</retrieval>
```

The properties in detail:

<search-filter>	A filter that matches every LDAP user object but nothing else. The format and interpretation of search-filter follows RFC 2254.
	
	Examples for search-filters:
	
	`(&(objectclass=user)(!(objectclass=computer)))`
	
	Match `objectclass=user`, but don't match `objectclass=computer`
	
	`(memberOf=cn=planforge,ou=groups,dc=ad,dc=example,dc=com)`
	
	Only match members of the group `"cn=planforge,ou=groups,dc=ad,dc=example,dc=com"`

<search-base>	The base to perform the search operation
<search-scope>	Either "object", "onelevel" or "subtree": • "object": Search of the base object only • "onelevel": Only return entries that are immediately below search-base • "subtree": Return entries on all levels from search base and below

The "<mapping>" sub-section

The <mapping> part of the <users> describes how LDAP user properties are mapped to Planforge properties. An example:

```
<mapping>
  <OpUser.Name value="sAMAccountName"/>
  <OpUser.Password value="userPassword"/>
  <OpUser.FirstName value="givenName"/>
  <OpUser.LastName value="sn"/>
  <OpUser.Email value="mail"/>
  <OpUser.Description value="description"/>
  <OpUser.Phone fixed="false" synched="true" value="telephoneNumber"/>
  <OpUser.Mobile fixed="false" synched="true" value="mobile"/>
  <OpUser.Fax fixed="false" synched="true" value="facsimileTelephoneNumber"/>
  <OpUser.GroupMembership value="memberOf"/>

  <OpUser.LinkedResource synched="false">
    <OpResource.Name value="sAMAccountName"/>
    <OpResource.FullName value="displayName"/>
    <!-- <OpResource.Number value="n/a"/> -->
    <OpResource.Description value="description"/>
  </OpUser.LinkedResource>
</mapping>
```

The `value`-attributes describe the fields within the LDAP user object.

User attributes that should be given a fixed value (not depending on any values within the LDAP user object) have set the `fixed="true"` attribute.

User attributes that should not be overwritten by subsequent update scheduler requests have a `synched ="false"` attribute.

User attributes might also be substituted based on Java regular expressions. As an example consider the following example which can be used to separate a retrieved "cn"-value to firstname and surname (Might be useful for OpenLDAP/ApacheDS):

```
<OpUser.FirstName value="cn">
    <replace>
        <from><![CDATA[(.*)[ ]+:sn$]]></from>
        <to><![CDATA[$1]]></to>
    </replace>
</OpUser.FirstName>
```

Your Planforge server's username will be set to the LDAP user property "cn", but the LDAP-user property "sn" will be removed. So if "cn" is "John W. Doe" and "sn" is "Doe" the Planforge user's first name will become "John W.".

Values within the "from" part followed after the ":" will be replaced with the first found value stored within the LDAP user object. In order to have a ":" within the from part please use "[:]".

$1,$2,.. values within the "to" part will be replaced with the first, second,.. matches within (..). If you are not familiar with the regular expressions, please make sure to follow this link:

http://docs.oracle.com/javase/7/docs/api/java/util/regex/Pattern.html.

⚠️

⚠ **Automatically setting the access level**

If you want the access level of users in Planforge to be set automatically depending on which groups they are members of in LDAP, an additional entry inside the <mapping> section is required. The relevant entry from the extended configuration example looks as follows:

```
<OpUser.Level value="memberOf" synched="true" default="observer">
  <replace>
    <from>CN=System,cn=planforge,dc=ad,dc=example,dc=com</from>
    <to>system</to>
  </replace>
  <replace>
    <from>CN=Manager,cn=planforge,dc=ad,dc=example,dc=com</from>
    <to>manager</to>
  </replace>
  <replace>
    <from>CN=Contributor,cn=planforge,dc=ad,dc=example,dc=com</from>
    <to>contributor</to>
  </replace>
  <replace>
    <from>CN=Observer,cn=planforge,dc=ad,dc=example,dc=com</from>
    <to>observer</to>
  </replace>
  <replace>
        <from>CN=Time-Tracking,cn=planforge,dc=ad,dc=example,dc=com</from>
        <to>time-tracking</to>
  </replace>
  <replace>
        <from>CN=External,cn=planforge,dc=ad,dc=example,dc=com</from>
        <to>external</to>
  </replace>
</OpUser.Level>
```

This example takes all "memberOf" values of users (group memberships) matching the <retrieval> sub-section above and checks each of these values. The replaces are executed in the same order as they are stated, and the first match returns the replacement value. If no match was found, then the "default" attribute is returned.

The <from> clause checks if a group matches the description, and if a match is found, then the value in the <to> clause is used to determine the access level.

If a user is a member of the Observer and the Contributor groups, then the mapping will check first the Manager section. If no group matches this section, so the groups will be checked against the Contributor section. Now a match is found (the Contributor group), so contributor is used as the access level mapping.

If the user is a member of the Managers group (note the additional s), or if the user does not belong to any group, then observer is used, and the user will be shadowed with the "Observer" access level in Planforge.

The "<OpUser.LinkedResource>" sub-section

If you want to automate resource creation for users then the most simple way is to add the following entry inside the <mapping> section:

```
<OpUser.LinkedResource synched="false">
  <OpResource.Name value="sAMAccountName"/>
  <OpResource.FullName value="displayName"/>
  <OpResource.Description value="description"/>
</OpUser.LinkedResource>
```

This entry automatically creates a resource for users newly synchronized from LDAP with a resource name equal to the user name, see the value="sAMAccountName" field. In general, if you want the user and resource to have the same name, please make sure that "OpResource.Name" and "OpUser.Name" have the same value="LdapAttribute" field.

The following example would prefix each resource with "Resource-" and then use the LDAP account name as resource name. This is not recommended but only used to show a possible configuration. The "Jira Username" of the resource will be set to "Jira- " and the user name (e.g. Jira-sAMAccountName) . The "Jira Username" property should normally not be needed. It is only relevant if you licensed the Jira Connector option and the user-names in Jira differ from your desired resource names.
The resource description would be of the form "Acct: account-name, descr: description-from-LDAP".

```
<OpUser.LinkedResource enabled="true" synched="true" autoCreate="true" assignToUserWhen="
resourceCreatedOrUnassigned">
  <OpResource.Name value="sAMAccountName" synched="false">
    <replace>
      <from>^(.*)$</from>
      <to>Resource-$1</to>
    </replace>
  </OpResource.Name>
  <!-- <OpResource.Number value="n/a" synched="false"/> -->
  <OpResource.JiraUsername value="sAMAccountName" synched="true">
    <replace>
      <from>^(.*)$</from>
      <to>Jira-$1</to>
    </replace>
  </OpResource.JiraUsername>
  <OpResource.FullName value="displayName" synched="true"/>
  <OpResource.Description value="sAMAccountName" synched="true">
    <replace>
      <from>^(.*)$</from>
      <to>Acct: $1, descr: :description</to>
    </replace>
  </OpResource.Description>
</OpUser.LinkedResource>
```

This specific example check the resources every time when a user is synchronized to Planforge and assigns a resource with matching name (in this case "Resource-username" to the user if the resource is newly created or if it is not yet assigned to a user.

Element description:

<OpUser.LinkedResource enabled="true">	Defines if resources are automatically linked/created (enabled="true") or if the whole <OpUser.LinkedResource/> section is ignored (enabled="false").
<OpUser.LinkedResource synched="true">	Defines if the resource will only be automatically created and/or assigned when creating a user in Planforge (synched="false") or each time a user logs in (synched="true"). Specifying synched="false" here overrides any synched="true" attributes of the child elements of <OpUser.LinkedResource>.
<OpUser.LinkedResource autoCreate="true">	Determines if resources are automatically created (autoCreate="true") if they do not yet exist.
<OpUser.LinkedResource assignToUserWhen="resourceCreatedOrUnassigned">	Handles how resources are assigned to users (the when part is influenced by "synched") Allowed values: • "never": resources are only created but never assigned to the user. This option was added for the sake of completeness only and is not recommended. • "resourceCreated": only resources which are created when the user is created are assigned. No already existing resource is touched • "resourceCreatedOrUnassigned": the resource is assigned to the user only if it is not already assigned to another user (so if the resource was created when the user logged in, or if was created earlier, but not assigned to any user) • "always": if the OpResource.Name entry produces a valid resource name, then the resource is assigned to the current user, even if it was already assigned to somebody else.
<OpResource.Name value="sAMAccountName" synched="false"/>	The value attribute defines which LDAP attribute is used for the new resource name.
<OpResource.Number value="n/a" synched="false"/>	The resource number.
<OpResource.JiraUsername value="sAMAccountName" synched="true"/>	The Jira Username attribute of the resource. Please note that this entry is only relevant if your license contains the Jira Connector option and if the resource name in Planforge is different from the user-names in Jira. More Information about this property can be found in Planforge's integrated documentation under Jira Connector > Resource synchronization.

<OpResource.FullName value=" displayName" synched="true" />	The full name (display name) of the resource.
<OpResource.Description value="description" synched=" true"/>	The description of the resource.
<replace>	This section can generally be omitted and should only be used when the desired resource property differs from the value provided by the LDAP "value=" attribute.
<from>	At least one mapping source value must match this element to be replaced with the <to> element. Elements are evaluated in the order in which they are stated, and the first matching one is returned. Please note that this section uses Java regular expressions to determine if a term matches. The "from" part may also contain placeholders for value attributes of the form ":attribute_name". For instance "<from>^(.*):sn$</from>" would return the part before the family name if value="cn" were configured.
<to>	A string representation of the desired Planforge resource property. This value can either reference the <from/> match (see the OpResource.Name example) or any other arbitrary value. Value attributes can also be referenced by stating them in the form ":attribute_name" (see the OpResource.FullName and OpResource.Description examples above).

⚠️ **Synchronize all users as "Deactivated" to Planforge**

If you want that all users will be synchronized as "Deactivated" to Planforge, an additional entry inside the <mapping> section is required. An example:

```
<ldap>
 <users>
   <mapping>
      <OpUser.Name value="sAMAccountName"/>
      <OpUser.Password value="userPassword"/>
           .......
      <OpUser.Active value="false" fixed="true" synched="false"/>
   </mapping>
 </users>
</ldap>
```

This example synchronizes all users as "Deactivated" to Planforge. Possible values for the entry <OpUser.Active value> are "true" or "false" (the default entry is true, because this field is optional). This value can also be set depending on LDAP fields and works similar to "Automatically setting the access level:". If you are using <replace> configuration entries, then note that the <to> part of the configuration must be <to>true</to> or <to>false</to>, and also do not forget to set the 'default="????"'-attribute of the <OpUser.Active .... > entry to "true" or "false".

6.8.4. Group Configuration ("<groups>")

The section is initiated with the `<groups>` tag and closes with `</groups>` - Here's an example:

```
<groups>
  <guid>objectGUID</guid>
    <retrieval>
      <search-filter>(objectClass=group)</search-filter>
      <search-base>ou=Groups,dc=ad,dc=example,dc=com</search-base>
      <search-scope>subtree</search-scope>
    </retrieval>
    <mapping>
      <OpGroup.Name value="sAMAccountName"/>
      <OpGroup.Description value="description"/>
      <OpGroup.ParentMembership value="memberOf"/>
    </mapping>
    <filter enabled="false">
      <include-names>.*ou=Groups,dc=ad,dc=example,dc=com</include-names>
    </filter>
</groups>
```

The "<guid>" unique identifier

<guid>	This field is required to track group name changes. Not all LDAP implementations offer a unique identifier for all entries inside the directory server, but if your LDAP server offers such an attribute then please make sure to specify it like shown in the example above

The "<retrieval>" sub-section

```
<retrieval>
    <search-filter>(objectClass=group)</search-filter>
    <search-base>ou=Groups,dc=ad,dc=example,dc=com</search-base>
    <search-scope>subtree</search-scope>
</retrieval>
```

All properties in detail:

<search-filter>	A filter that matches every LDAP group object but nothing else. The format and interpretation of search-filter follows RFC 2254.
<search-base>	The base to perform the search operation
<search-scope>	Either "object", "onelevel" or "subtree": • "object": Search of the base object only • "onelevel": Only return entries that are immediately below search-base • "subtree": Return entries on all levels from search base and below

The "<mapping>" sub-section

This section describes how LDAP group properties are mapped to Planforge properties.

```
<mapping>
   <OpGroup.Name value="sAMAccountName"/>
   <OpGroup.Description value="description"/>
   <OpGroup.ParentMembership value="memberOf"/>
</mapping>
```

The `value` attributes describe the fields within the LDAP group object.

Group attributes that should be given a fixed value (not depending on any values within the LDAP group object) have set the `fixed="true"` attribute.

Group attributes that should not be overwritten by subsequent update scheduler requests have a synched="false" attribute.

Group attributes might also be substituted based on Java regular expressions. It works the same way like shown in the exmaple for the `<users>` mapping.

Group filters

Group filters are used to control which LDAP groups shall be shadowed to the Planforge server's database:

```
<filter enabled="false">
   <include-names>.*ou=Groups,dc=ad,dc=example,dc=com</include-names>
</filter>
```

The filtering mechanism is enabled by setting `enabled="true"`. if filtering is disabled (by setting enabled="false") all LDAP groups under the given group search-base will be shadowed to the Planforge database. In detail:

<include-names>	The filter criteria - for the example above all subsequent groups with a credential ending like "ou=Groups,dc=ad,dc=example,dc=com" are matched and therfore would be shadowed to the Planforge server's database.
	There may be arbitrary many "`<include-names></include names>`" attributes. If an LDAP group matches one of them it will be shadowed.

Testing the configuration

After the LDAP-configuration was applied to the file `"configuration.oxc.xml"` and the Tomcat application-server was restarted, your LDAP-configuration will be active.

After a LDAP user-account signs on for the first time, this LDAP-enabled usser-account will be stored in the Planforge server's database.

If you want to manually force a full sychronization-cycle to retrieve all LDAP-objects that can be reached by your configuration:

1. Sign on as "administrator" at the login-form of your Planforge server
2. Open the system settings by clicking the tool-shaped button in the upper-right corner
3. Click the tool "Users" in the tool-dock on the left side
4. Click the button "Synchronize"

> ⚠️ **LDAP Troubleshooting**
>
> If your LDAP-configuration doesn't seem to work:
>
> - Check Tomcat's main-logfile for warnings or error-messages
> - Test the configuration-properties you have entered in the LDAP-configuration's `<conn ection>` section with an LDAP-browser

> 🛑 **Anonymous Binds**
>
> After an LDAP-connection was activated please make sure to check if you are allowed to sign on to your Planforge server without specifying a password (Or with a wrong password) - If this is the case then most likely **Anonymous Bind** is allowed by your LDAP-server and **must be disabled**!

6.9. Active Directory Single Sign-On Configuration

Element description: "Single Sign-on"-functionality and "Multi-Factor Authentication" can be activated by connecting Planforge to an "Azure Active Directory" cloud authority, an "Azure AD B2C" authority or an "Active Directory Federation Services" (AD FS) authority (AD FS 2019 and above). Logging in to Planforge will be accomplished via the "Active Directory Login" web-page . To enable this functionality follow these steps:

1. Connect to your Azure AD or AD FS portal.
2. Register Planforge as a Web application as described in one of the following documents:
    1. Azure AD: https://docs.microsoft.com/en-us/azure/active-directory/develop/quickstart-register-app
    2. AD FS: https://docs.microsoft.com/en-us/windows-server/identity/ad-fs/development/msal/adfs-msal-web-app-web-api
3. Allow users to access the Planforge application:
    1. Azure AD: https://docs.microsoft.com/en-us/azure/active-directory/manage-apps/add-application-portal-assign-users
    2. AD FS: user access can be configured under the "Apply Access Control Policy" screen of the "Add Application Group Wizard" dialog.
4. In the "Manage" section of the web application, item: "Authentication", configure the redirect URI and make sure that "Access tokens" and "ID tokens" are enabled in the "Implicit grant and hybrid flows" of this page.
5. This step is optional, but recommended since the following attributes can not be synchronized to Planforge otherwise:
   In the "Manage" section, item: "Token configuration" make sure that the "Optional claims" section contains the claims "email", "family_name", "given_name", "preferred_username" and "xms_pl". These claims must be configured for token type "ID".
   A notable exception is the claim "name" (the full name of the user) which can not be configured, however it seems to be part of the "ID" token anyway.
6. Navigate to the "Overview" tab and use the value of:
    1. "Directory (tenant) ID" to replace the the "__tenant-id__" string in the "Active Directory SSO Connection Parameters": "active-directory-sso/connection/authority" section in the configuration of Planforge as seen below and the example configurations. This section should now look like:

       > *<authority>https://login.microsoftonline.com/xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx</authority>*

    2. "Application (client) ID" in the "Active Directory SSO Connection Parameters": "active-directory-sso/connection/application-id" section in the configuration file of Planforge.
7. Make sure that the "active-directory-sso/connection/redirect-uri" in the configuration file is the same as configured under step 4.
8. In the "Manage" section of the web application, item: "Certificates & secrets", create a client secret and use this value in the "active-directory-sso/connection/client-secret" section in the configuration file.
9. Optionally, adjust the "User Configuration" parameters to adapt user creation to your needs.

⚠

> ⚠️ **User access limitations and details**
>
> - Azure AD and AD FS SSO users will only exist in Planforge after they log in to Planforge the first time, they can not be pre-created by an administrator.
> - If a user tries to log in to Planforge with an account whose user-name is already present in the system (created via LDAP or by an administrator) then the user will get an "Duplicate user or group name" error.
> - If an SSO user tries to log in via the Planforge login screen, the user will get a "The user name is unknown or the password you entered is not correct" error.
> - If a user could log in via SSO but is disabled in Planforge then the user will get a "The user name is unknown or the password you entered is not correct" error.
> - Although users can be disabled in Planforge's user management, the preferred way to handle access to Planforge is via the Azure AD and AD FS user access dialogs.
> - It is not recommended to configure "active-directory-sso" and LDAP or Atlassian Crowd authentication at the same time. Configuring SSO together with the other authentication services can be enabled by replacing the first line of the SSO configuration <active-directory-sso> with:
>   <active-directory-sso allowMultipleExternalUserDirectories="true">

6.9.1. Active Directory SSO Authentication

The SSO configuration can be fine-tuned to meet your specific needs, however, the minimal configuration should be acceptable for most use-cases.

> ⚠️ **Example Files**
>
> Following this guide will be easier with an example configuration. You will find three configuration-files inside the folder "adssoconfigs" of the downloaded "Planforge" zip-archive:
>
> - "minimal" contains a minimal configuration which hides all <user> configuration entries. This configuration behaves exactly the same way as the "standard" configuration.
> - "standard" explicitly shows the configuration as used for the "minimal" case.
> - "extended" is similar to "standard" but shows how to use regular expressions to extract only the name-part of the preferred_username (if the preferred_username is an email-address, otherwise the whole name is used)

All Active Directory SSO configuration-properties will be entered in between the tags `<active-directory-sso>` and `</active-directory-sso>`, consisting of two sections:

```
    </database>

  <active-directory-sso>
    <connection>
        <!-- Contains the Active Directory SSO connection parameters -->
    </connection>
    <users>
      <!-- OPTIONAL - Configures which user properties-will be retrieved
           and how their access level will be mapped to Planforge -->
    </users>
  </active-directory-sso>

</configuration>
```

The section `<connection>` is mandatory.
Leaving out the <users> section will work the same as if the "standard" configuration (see section: "User Configuration" below) was configured.

The following sections describe the Active Directory SSO configuration in detail.

6.9.2. Active Directory SSO Connection Parameters ("<connection>")

```
<connection>
  <authority>https://login.microsoftonline.com/__tenant-id__</authority>
  <application-id>yyyyyyyy-yyyy-yyyy-yyyy-yyyyyyyyyyyy</application-id>
  <client-secret>the-application-specific-client-secret-string</client-secret>
  <redirect-uri>https://my.planforge.server/planforge</redirect-uri>
</connection>
```

All connection properties in detail:

<authority>	The URL where the single-sign-on token can be requested from. This URL usually has the form "https://login.microsoftonline.com/xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx" where the "xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx" part represents the tenant-ID (see https://docs.microsoft.com/en-us/azure/active-directory/fundamentals/active-directory-how-to-find-tenant for how to find this ID).
<application-id>	The ID under which Planforge was registered in Azure AD or AD FS (see https://docs.microsoft.com/en-us/azure/active-directory/develop/howto-create-service-principal-portal for more details).
<client-secret>	The application secret used by Planforge which was created in Azure AD or AD FS (see https://docs.microsoft.com/en-us/azure/active-directory/develop/howto-create-service-principal-portal for more details).
<redirect-uri>	The URI where the access token is sent, this is also the URI for Planforge (see https://docs.microsoft.com/en-us/azure/active-directory/develop/howto-create-service-principal-portal for more details).

6.9.3. User Configuration ("<users>")

Before you continue with this section, please note that Planforge is licensed on a per-user basis and each user account synchronized from Azure AD or AD FS represents one licensed user. To check how many user accounts can still be created inside your Planforge server:

1. Sign on as a user with the access level "System" at the login-form of your Planforge server
2. Open the system settings by clicking the tool-shaped button in the upper-right corner
3. Click the tool "License" in the tool-dock on the left side

The lines "Number of Observer Users", "Number of Contributor Users" and "Number of Manager Users" will show in brackets the maximum count of user-accounts for each user-level. The number on the left will show how many user-accounts have already been created for each user-level.

Azure AD or AD FS users are synchronized to Planforge when they log in to Planforge. Consequently, they are only created in Planforge when they log in to it for the first time, and from this time on they count as licensed users. Even if the users can be disabled in Planforge (and thus no longer count as licensed users), these users will get potentially confusing error messages when trying to log in to Planforge. It is thus highly recommended that you also configure in Azure AD / AD FS which users may access the Planforge application. Please note that in the default configuration, users are created as "Manager Users" in Planforge if no roles are defined in your Azure AD / AD FS app configuration. This could be changed by replacing the default="manager" attribute with e.g. default="observer" in the OpUser.Level section. If you want to manage the user level within Planforge, please change <OpUser.Level value="roles" default="manager" synched="true"/> to <OpUser.Level value="roles" default="manager" synched="false" />

The section describing the user-retrieval and access level mapping is initiated with the `<users>` tag and closes with `</users>` - This example works the same as if the <users> section were omitted:

```
<users>
    <OpUser.Name value="preferred_username" synched="true"/>
    <OpUser.FirstName value="given_name" synched="true"/>
    <OpUser.LastName value="family_name" synched="true"/>
    <OpUser.Email value="email" synched="true"/>
    <!-- <OpUser.Description value="n/a" synched="false"/> -->
    <OpUser.Language value="xms_pl" synched="false"/>
    <OpUser.Level value="roles" default="manager" synched="true"/>
    <!-- <OpUser.Phone value="n/a" synched="false"/> -->
    <!-- <OpUser.Mobile value="n/a" synched="false"/> -->
    <!-- <OpUser.Fax value="n/a" synched="false"/> -->
    <OpUser.Active value="true" fixed="true" synched="false"/>
    <OpUser.LinkedResource enabled="true" synched="false" autoCreate="true" assignToUserWhen="
resourceCreatedOrUnassigned">
        <OpResource.Name value="preferred_username" synched="false"/>
        <!-- <OpResource.Number value="n/a" synched="false"/> -->
        <OpResource.FullName value="name" synched="true"/>
        <!-- <OpResource.Description value="n/a" synched="false"/> -->
    </OpUser.LinkedResource>
</users>
```

The elements in the value="XXXX" attributes are claims in the ID token returned by Azure AD / AD FS. These (and additional) claims can be configured as described in https://docs.microsoft.com/en-us/azure/active-directory/develop/active-directory-optional-claims
The ID token can currently not be configured to contain meaningful values for the commented-out user attributes (OpUser.Description, Phone, Mobile and Fax). However, these attribute mappings are implemented and can be configured as soon as the claim set for the ID token is expanded by Microsoft.

⚠️

> ⚠️ **Claims contained in the ID token**
>
> To see which claims are available in the ID token, add the following lines to your "log4j.properties" file.
>
> ```
> # specific active-directory-sso claims
> log4j.logger.active-directory-sso.ClaimsLogger=TRACE, opp, oppf
> log4j.additivity.active-directory-sso.ClaimsLogger=false
> ```
>
> If a user logs in to Planforge via Azure AD or AD FS SSO then the planforge.log file will contain the following info:
>
> ```
> ************** claims begin ********************
> * 'email' = 'cs-email@planforge.io'
> * 'exp' = 'Fri Mar 05 10:11:28 GMT 2021'
> * 'family_name' = 'Schulz'
> * 'given_name' = 'Claudia'
> * 'name' = 'Claudia Schulz'
> * 'preferred_username' = 'cs@planforge.io'
> * 'roles' = '["manager"]'
> ...
> * 'ver' = '2.0'
> * 'xms_pl' = 'de-DE'
> ************** claims end *******************
> ```
>
> All keys in this list (like 'email' without the single quotes) can be used as a value in the user configuration.

The "<OpUser.Level>" sub-section

If you want the access level of users in Planforge to be set automatically, you have to create roles in the Azure AD / AD FS app registration and assign these roles to users (see https://docs.microsoft.com/en-us/azure/active-directory/develop/howto-add-app-roles-in-azure-ad-apps for more details). The value of the role must be exactly the same as the recognized values stated in the "<to>" section below. Please note that the default configuration currently only supports the assignment of one role. If more than one role is assigned in Azure AD / AD FS then one is selected randomly.
To allow the assignment of multiple roles in Azure AD (and also to allow custom values for roles) please use a configuration similar to the following section. This example basically uses the first matching role as the result.
This would also allow the mapping of groups to user levels by changing the value="roles" attribute in the example below to value="groups" and by adjusting the <from> sections accordingly. However, bear in mind that only group IDs are returned in the ID token from Azure AD (see https://docs.microsoft.com/en-us/azure/active-directory/develop/active-directory-optional-claims#configuring-directory-extension-optional-claims for more details).

```
<OpUser.Level value="roles" synched="true" default="observer">
    <replace>
        <from>^system$</from>
        <to>system</to>
    </replace>
    <replace>
        <from>^manager$</from>
        <to>manager</to>
    </replace>
    <replace>
        <from>^time-tracking$</from>
        <to>time-tracking</to>
    </replace>
    <replace>
        <from>^contributor$</from>
        <to>contributor</to>
    </replace>
    <replace>
        <from>^external-contributor$</from>
        <to>external-contributor</to>
    </replace>
    <replace>
        <from>^observer$</from>
        <to>observer</to>
    </replace>
    <replace>
        <from>^customer$</from>
        <to>customer</to>
    </replace>
    <replace>
        <from>^external$</from>
        <to>external</to>
    </replace>
    <replace>
        <from>^special-customer-defined-manager-role$</from>
        <to>manager</to>
    </replace>
</OpUser.Level>
```

> ⚠ **User-level mapping syntax considerations**
>
> Please note that the <from> part uses Java regular expressions, specifically:
>
> - if you want an exact string match (in contrast to a sub-string match) then the matching term has to be surrounded by ^ and $ (as in the examples).
> - one-character wildcard is a colon (".")
> - multi-character wildcard is colon-asterisk (".*").
>
> For more details please see: http://docs.oracle.com/javase/7/docs/api/java/util/regex/Pattern.html

Element description:

`<OpUser.Level value=" roles">`	The source attribute of the user level mapping defines where the data in the \<from\> element of the \<replace\> section is taken from. Recommended values are: • "roles" maps from app roles as defined in Azure AD / AD FS to Planforge roles. • "groups" which performs the mapping based on group membership.
`<OpUser.Level synched=" true">`	Defines if the user level will be maintained in Azure AD / AD FS (synched="true") or only set when creating the user in Planforge (synched="false"). Please note that the user levels maintained externally can not be changed in Planforge (will be read-only in the "Edit User" dialog).
`<OpUser.Level default=" observer" >`	The default user level if the user level source element does not match any element in the specified \<from\> sections. Recognized values are the same as in the \<to\> section described below.
\<from\>	At least one mapping source value must match this element to be replaced with the \<to\> element. Elements are evaluated in the order in which they are stated, and the first matching one is returned. In the above example, if the user is assigned to the "system" and "external" roles in Azure AD / AD FS, then "system" would be returned. This example basically ensures that the user always has the maximum specified level. Please note that this section uses Java regular expressions to determine if a term matches.
\<to\>	A string representation of the desired Planforge user level. Recognized values are: "system", "manager", "contributor", "external-contributor", "time-tracking", "observer", "customer", and "external"

The "\<OpUser.LinkedResource\>" sub-section

The synchronization of users between Azure AD / AD FS and Planforge only happens when a user logs in, and pre-sync or pre-creation of the user in Planforge is currently not supported. To simplify the resource creation and administration in Planforge, resources are created and assigned to users (if not already assigned) by default. To disable automatic resource creation altogether, please add the following configuration section to your configuration.oxc.xml file in the "active-directory-sso"/"users" section:

```
<OpUser.LinkedResource enabled="false"/>
```

The following example would create a resource with the name-part (everything before the "@") of the preferred user name, or the whole preferred user name if no "@" were present in it or if it started wit an "@". Removing the second <from> section will prevent resource-creation and assignment if the preferred_username does not contain an "@" (the resource name would be empty and thus no resource would be created).

The second part sets the "Jira Username" of the resource to "Jira- " and the resource name (e.g. Jira-EmailNamePart) . The "Jira Username" property should normally not be needed. It is only relevant if you licensed the Jira Connector option and the user-names in Jira differ from your desired resource names.

The third part sets the full name of the resource to the full name provided by Azure AD / AD FS.

The description of the resource will be something like "This is Doe, Jane with preferred language: en-US"

```
    <OpUser.LinkedResource enabled="true" synched="true" autoCreate="true" assignToUserWhen="
resourceCreatedOrUnassigned">
        <OpResource.Name value="preferred_username" synched="false">
            <replace>
                <from>^([^@]+)@.*$</from>
                <to>$1</to>
            </replace>
            <replace>
                <from>^(.*)$</from>
                <to>$1</to>
            </replace>
        </OpResource.Name>
        <!-- <OpResource.Number value="n/a" synched="false"/> -->
        <OpResource.JiraUsername value="preferred_username" synched="true">
            <replace>
                <from>^([^@]*).*$</from>
                <to>Jira-$1</to>
            </replace>
        </OpResource.JiraUsername>
        <OpResource.FullName value="name" synched="true"/>
        <OpResource.Description value="preferred_username" synched="true">
            <replace>
              <from>^.*$</from>
              <to>This is :family_name, :given_name with preferred language: :xms_pl</to>
            </replace>
        </OpResource.Description>
    </OpUser.LinkedResource>
```

Element description:

\<OpUser. LinkedResource enabled=" true">	Defines if resources are automatically linked/created (enabled=" true") or if the whole \<OpUser.LinkedResource/> section is ignored (enabled="false"). Please note that resources are created by default (see beginning of this section).
\<OpUser.LinkedResource synched="true">	Defines if the resource will only be automatically created and/or assigned when creating a user in Planforge (synched="false") or each time a user logs in (synched="true"). Specifying synched=" false" here overrides any synched="true" attributes of the child elements of \<OpUser.LinkedResource>.
\<OpUser. LinkedResource autoCreate=" true">	Determines if resources are automatically created (autoCreate=" true") if they do not yet exist.
\<OpUser.LinkedResource assignToUserWhen=" resourceCreatedOrUnassigned" >	Handles how resources are assigned to users (the when part is influenced by "synched") Allowed values: • "never": resources are only created but never assigned to the user. This option was added for the sake of completeness only and is not recommended. • "resourceCreated": only resources which are created when the user is created are assigned. No already existing resource is touched • "resourceCreatedOrUnassigned": the resource is assigned to the user only if it is not already assigned to another user (so if the resource was created when the user logged in, or if was created earlier, but not assigned to any user) • "always": if the OpResource.Name entry produces a valid resource name, then the resource is assigned to the current user, even if it was already assigned to somebody else.
\<OpResource.Name value=" preferred_username" synched="false"/>	The value attribute defines which Azure AD / AD FS attribute or user detail is used for the new resource name. Sensible values are: • "preferred_username": the user name retrieved from Azure AD / AD FS. • "email": the user's email address. • "name": the name (given_name + family_name) of the user.
\<OpResource.Number value="n /a" synched="false"/>	The resource number.

<OpResource.JiraUsername value="preferred_username" synched="true">	The Jira Username attribute of the resource. Sensible value are the same as the ones specified in "OpResource.Name". Please note that this entry is only relevant if your license contains the Jira Connector option and if the resource-name in Planforge is different from the user names in Jira. More Information about this property can be found in Planforge's integrated documentation under Jira Connector > Resource synchronization.
<OpResource.FullName value=" name" synched="true">	The full name (display name) of the resource.
<OpResource.Description value="n/a" synched="false"/>	The description of the resource.
<replace>	This section can generally be omitted and should only be used when the desired resource property differs from the value provided by the Azure AD / AD FS "value=" attribute.
<from>	At least one mapping source value must match this element to be replaced with the <to> element. Elements are evaluated in the order in which they are stated, and the first matching one is returned. Please note that this section uses Java regular expressions to determine if a term matches. The "from" part may also contain placeholders for value attributes of the form ":attribute_name". For instance "<from>^(.*):family_name$</from>" would return the part before the family name if value="name" were configured.
<to>	A string representation of the desired Planforge resource property. This value can either reference the <from/> match (see the OpResource.Name example) or any other arbitrary value. Value attributes can also be referenced by stating them in the form ":attribute_name" (see the OpResource.Name and OpResource. Description examples above).

6.10. Atlassian Crowd Integration

A "Single Sign-on"-functionality is available by linking Planforge to "Atlassian Crowd". Logging in to Crowd will also log you in to Planforge with the corresponding user and vice versa. To enable this functionality follow these steps:

1. Install and configure Atlassian Crowd as documented here: https://confluence.atlassian.com/display/CROWD/Installing+Crowd
2. Set up a new custom application within Crowd as described here: https://confluence.atlassian.com/display/CROWD/Integrating+Crowd+with+a+Custom+Application
   Name the application 'planforge', in the below example 'secret' was chosen as password. Make sure to configure the 'Remote addresses' to match your Planforge server address.
   Though not strictly necessary, it is advisable to configure a group in Crowd which can authenticate to the Planforge application.
3. Configure Users: Make sure the users are part of the group configured in the step above.

> ⚠ **Testing and domain considerations**
>
> Please note that Atlassian Crowd adds a special cookie to your Planforge session, this cookie has a Domain=xxx.yyy part which causes your browser to ignore the cookie when Planforge and Crowd are in different domains.
>
> This should not be a problem in production environments, however it might be an issue if a test environment tries to authenticate with a production Crowd instance in a different domain.

6.10.1. Crowd Authentication

Though Atlassian Crowd can be used with a minimal configuration of the connection parameters only, it is usually desirable to fine-tune the user and group synchronization process.

> ⚠ **Example Files**
>
> Following this guide will be easier with an example configuration. You will find six example configuration-files inside the folder "crowdconfigs" of the downloaded Planforge zip-archive:
>
> - "minimal" contains a minimal configuration which should help getting started: all Crowd users and no groups are synchronized to Planforge.
> - "groupsnested" synchronizes active users and groups. This example assumes that nested groups are enabled on the Crowd directory and that the PLANFORGE group is the parent group of all users and groups which should be synchronized to PLANFORGE
> - "groupsnonnested" also synchronizes active users and groups. This example can be used if you did not enable nested groups in Crowd. The users that you would like to synchronize must be members of the PLANFORGE
> - group, and the groups you would like to synchronize must be named "PLANFORGE" or start with PF_ (case-sensitive).
> - "userlevelbygroup" adds user level mapping based on group relationship. The user level is set only when the user is first created in Planforge (synched="false') and can be changed by users with the access level "System".

- "userlevelbyattribute" performs user level mapping based on the custom attribute "pfUserLevel". The user level is maintained by the Crowd attribute (synched="true) and cannot be modified in Planforge.
- "extended": a not really practical example showing mainly what configuration parameters would be possible.

All Crowd configuration-properties will be entered inbetween the tags `<crowd>` and `</crowd>`, consisting of four main-sections:

```
  </database>

  <crowd>
    <connection>
       <!-- Contains the Crowd-server's connection parameters -->
    </connection>
    <update-schedule>
      <!-- OPTIONAL - Configures how often the Crowd-synchronization
           cycle will be initialized -->
    </update-schedule>
    <users>
      <!-- OPTIONAL - Configures which user-accounts will be retrieved
           and how their access level will be mapped to Planforge -->
    </users>
    <groups>
      <!-- OPTIONAL - Configures which groups will be retrieved
           from the directory server and how their attributes
           will be mapped to  Planforge -->
    </groups>
  </crowd>

</configuration>
```

The section `<connection>` is mandatory, the other sections have the following default behavior:

- Leaving out <users> will synchronize all Crowd users to Planforge (normally used for testing purposes only)
- Omitting <schedule> will disable periodic synchronization (users are updated whenever they log in to Planforge)
- Leaving out the section `<groups>` will prevent Crowd groups from being created in Planforge.

The following sections describe the Crowd-configuration in detail.

6.10.2. Crowd Connection Parameters ("<connection>")

```
<connection>
  <application-name>planforge</application-name>
  <application-password encrypted="false">secret</application-password>
  <crowd-server-url>http://my.crowd.server:8095/crowd</crowd-server-url>
  <!-- <session-validationinterval>0</session-validationinterval> -->
</connection>
```

All connection properties in detail:

<application-name>	The name of the Planforge application as specified when setting up the application in Crowd.
<application-password encrypted="false">	The password used for the Planforge application in Crowd. The entered information will be encrypted during the next start of Planforge and the entry will be set to "<security-credentials encrypted="true">"
<crowd-server-url>	The URL to connect to the Atlassian Crowd server. The URL has to start with either "http://" or "https://". Please do not forget to specify the port (usually 8095)
<session-validationinterval>	The number of minutes to cache authentication validation in the session. If this value is set to `0`, each HTTP request will be authenticated with the Crowd server.

6.10.3. Update Scheduler settings ("<update-schedule>")

These settings are used to control the periodical updates. These updates will synchronize any changes (if configured) from the Crowd server to the Planforge database. Typical updates are: every day at 5 AM from Monday to Friday and this is exactly what the following example configuration does:

```
<update-schedule syncNewUsers="true">
  <minute>0</minute>
  <hour>5</hour>
  <month>*</month>
  <day-of-week>1-5</day-of-week>
 <!-- or: <day-of-month>1,10,20</day-of-month> -->
</update-schedule>
```

⚠

> ⚠ **Synchronizing new created users from Crowd to Planforge**
>
> For a successful synchronization of new created users from Crowd to Planforge, the attribute "syncNewUsers" has been implemented. This attribute is optional and possible values for this attribute are "true" and "false" (default = "false"):
>
> 1. If set to "false" or not specified, then users newly created in Crowd are not automatically synchronized to Planforge. These users will be synchronized if the administrator opens the "Users"-tool in the System Settings and presses the "Synchronize"-button there. If a non-synchronized user logs in to Planforge, then this user is automatically synchronized. Setting the syncNewUsers attribute to "false" makes the administrator more aware when new users are imported and also prevents Planforge from automatically using licenses for new LDAP users.
> 2. If set to "true", then all users matching the <users><filter> section (see below) will be automatically synchronized to Planforge. Setting this attribute to "true" allows resource managers to link resources to new users without having to ask the administrator to import these new users.

The allowed values in detail:

<minute>	0-59	, - * /
<hour>	0-23	, - * /
<month>	1-12 or JAN-DEC	, - * /
<day-of-week>	1-7 or SUN-SAT	, - * / L #
<day-of-month>	1-31	, - * / L W

Either day-of-week or day-of-month must be specified. The values/special-characters that can be entered are actually separate fields of a "cron Expression". You can find the full description to these fields at: http://www.quartz-scheduler.org/docs/tutorials/crontrigger.html

6.10.4. User Configuration ("<users>")

Before you continue with this section please note that Planforge is licensed on a per-user basis and each user-account synchronized from Crowd represents one licensed user. To check how many user-accounts can still be created inside your Planforge server:

1. Sign on as "administrator" at the login-form of your Planforge server
2. Open the system settings by clicking the tool-shaped button in the upper-right corner
3. Click the tool "License" in the tool-dock on the left side

The lines "Number of Observer Users", "Number of Contributor Users" and "Number of Manager Users" will show in brackets the maximum count of user-accounts for each user-level. The number on the left will show how many user-accounts have already been created for each user-level.

Even if Crowd groups which may access the Planforge application are configured in Atlassian Crowd, these groups cannot be determined automatically by Planforge. It is thus highly recommended that you specify these groups in a <parent-group> filter to limit which users are synchronized to Planforge.

The section describing the user-retrieval and access level mapping is initiated with the `<users>` tag and closes with `</users>` - Here's an example what this section could look like:

```
<users>
  <filter mode="all">
    <parent-group>PLANFORGE</parent-group>
    <name>PF*</name>
    <email>*@planforge.io</email>
    <displayName>* C?O of *</displayName>
    <active>true</active>
    <attribute>pfAttribute=allow</attribute>
  </filter>
  <OpUser.Active value="active" synched="false"/>
  <OpUser.Level value="group" synched="false" default="external">
    <replace>
      <from>^PF_System$</from>
      <to>system</to>
    </replace>
    <replace>
      <from>^PF_Manager$</from>
      <to>manager</to>
    </replace>
    <replace>
      <from>^PF_Observer$</from>
      <to>observer</to>
    </replace>
  </OpUser.Level>
  <!-- Optional automatically create resources linked to users
  <OpUser.LinkedResource synched="false">
      <OpResource.Name value="name"/>
      <OpResource.FullName value="displayName"/>
  </OpUser.LinkedResource>     -->
</users>
```

The "<filter>" sub-section

This sub-section inside `<users>` defines which user-accounts shall be synchronized to Planforge, an example:

```
<filter mode="all">
  <parent-group>PLANFORGE</parent-group>
  <active>true</active>
</filter>
```

> ⚠️ **Managing user access to Planforge**
>
> Please note that the <filter> section of the <users> configuration only restricts access to Planforge. If the user is not in a group which may access the Planforge application then the user will not able to log in even if it is as an active user in "System/ADMINISTRATE/Users". If this filter is missing or contains no entries, then all Crowd users will be synchronized to Planforge (even users who may not log in to the Planforge application).

Filters are case-sensitive and support partial matches (via *) and single-character wild-cards (via ?). If the filtered attribute has more than one value in Crowd, then all values are checked and at least one element must match the configured filter element to satisfy the criteria. All filter sub-sections, with the exception of <active> can be specified multiple times.

The elements in detail:

<filter mode="all">	The filter mode can be "all" or "any", all means that all specified filter terms must match, and "any" means that the user is synchronized as soon as one of the specified filer items matches. The above example, for instance, states that the users must be in the group "PLANFORGE", must be active in Crowd, and must have a user-defined attribute called "pfAttribute" with the value "allow"
<parent-group>	A group of which the user should be a (possibly nested) member.
<name>	The user name.
<active>	If the user is active or inactive.
<email>	The user's email address.
<firstName>	The first name of the user.
<lastName>	The last name of the user.
<displayName>	The display name of the user.
<attribute>	A user defined attribute in the form attribute=value. With "attribute" being the name of the user-defined attribute in Crowd and "value" being the term to filter.

The "<OpUser.Active>" entry

This entry defines if and how the "active" flag of users in Planforge is synchronized when importing or updating users from Atlassian Crowd

If your configuration does not contain an <OpUser.Active> entry then synchronization of the "active" flag works as if the following entry were specified:

```
<OpUser.Active value="active" synched="true"/>
```

Another common configuration entry will be:

```
<OpUser.Active value="true" synched="false" fixed="true"/>
```

Element description:

<OpUser. Active value=" active">	The "value" attribute of the user mapping defines what element of the Atlassian Crows user determines the "active" value. normally only "active" should be needed. However, to if you need to implement a sophisticated mapping then you can use any of values and elements described in the <OpUser.Level> section below. The value must return the string "true" if the synchronized user should be active. This can also be achieved by specifying one or more <replace> sections. In addition to the values allowed for <OpUser.Level> all Atlassian Crowd user properties (like "name", "active", ... see the filter description for allowed values) can be used.
<OpUser. Active synched=" true">	Defines if the active flag be maintained in Crowd (synched="true") or only set when creating the user in Planforge (synched="false"). Please note that the active flags maintained by Crowd can not be changed in Planforge (will be read-only in the "Edit User" dialog).
<OpUser. Active value=" true" synched=" false" fixed=" true"/>	The fixed="true" entry indicates that the value is not a property or attribute but is taken as specified, and is normally only used together with a synched="false" entry. This example sets the "active" flag of Planforge users to true if they are synchronized the first time and does not update the flag regardless of the "active" flag in Atlassian Crowd. The synched="false" part allows the administrator to edit the "active" status in the Planforge user management interface.

The "<OpUser.Level>" sub-section

If you want the access level of users in Planforge to be set automatically depending on group membership or a user-defined attribute, an additional entry inside the <users> section is required. An example:

```
<OpUser.Level value="attribute:pfUserLevel" synched="true" default="external">
    <replace>
        <from>^pf-system$</from>
        <to>system</to>
    </replace>
    <replace>
        <from>^pf-manager$</from>
        <to>manager</to>
    </replace>
    <replace>
        <from>^pf-time-tracking$</from>
        <to>time-tracking</to>
    </replace>
    <replace>
        <from>^pf-contributor$</from>
        <to>contributor</to>
    </replace>
    <replace>
        <from>^pf-external-contributor$</from>
        <to>external-contributor</to>
    </replace>
    <replace>
        <from>^pf-time-tracking$</from>
        <to>time-tracking</to>
    </replace>
    <replace>
        <from>^pf-observer$</from>
        <to>observer</to>
    </replace>
    <replace>
        <from>^pf-customer$</from>
        <to>customer</to>
    </replace>
    <replace>
        <from>^pf-external$</from>
        <to>external</to>
    </replace>
</OpUser.Level>
```

This example takes the user-defined Crowd attribute "pfUserLevel" of a Crowd user and converts it into a user-level understood by Planforge. The user-levels are maintained in Crowd and can thus not be edited in Planforge.

Please note the differences to the example shown in the beginning of this chapter, specifically value=" group" and synched="false". The term value="group" means that (nested) group membership of users will determine the user-level. synched="false" states that the user-level is maintained in Planforge and can thus be edited in the "Edit User" dialog in "System/ADMINISTRATE/ Users".

However as a mapping is defined, the user-level is set when the system creates the user in Planforge (the first time the user logs in to Planforge, or when the synchronization process encounters a new user which may be synchronized to Planforge).

⚠

> ⚠️ **User-level mapping syntax considerations**
>
> Please note that the <from> part uses Java regular expressions, specifically:
>
> - if you want an exact string match (in contrast to a sub-string match) then the matching term has to be surrounded by ^ and $ (as in the examples).
> - one-character wildcard is a colon (".")
> - multi-character wildcard is colon-asterisk (".*").
>
> For more details please see: http://docs.oracle.com/javase/7/docs/api/java/util/regex/Pattern.html

Element description:

<OpUser. Level value=" attribute: pfUserLevel" >	The "value" attribute of the user level mapping defines where the data in the <from> element of the <replace> section is taken from. Allowed values are: - "group" which performs the mapping based on (nested) group membership. - "attribute:crowdUserLevelAttributeName" which indicates that a user defined Crowd attribute with the specified name (in this example "pfUserLevel") is the source of the user level mapping.
<OpUser. Level synched=" true">	Defines if the user level will be maintained in Crowd (synched="true") or only set when creating the user in Planforge (synched="false"). Please note that the user levels maintained by Crowd can not be changed in Planforge (will be read-only in the "Edit User" dialog).
<OpUser. Level default=" observer">	The default user level if the user level source element does not match any element in the specified <from> sections. Recognized values are the same as in the <to> section described below.
<from>	At least one mapping source value must match this element to be replaced with the <to> element. Elements are evaluated in the order in which they are stated, and the first matching one is returned. In the above example, if "pfUserLevel" had the values "pf-system"and "pf-external" in Crowd (Crowd allows multi-value user defined attributes), then "system" would be returned. This example basically ensures that the user always has the maximum specified level. Please note that this section uses Java regular expressions to determine if a term matches.
<to>	A string representation of the desired Planforge user level. Recognized values are: "system", "manager", "contributor", "external-contributor", "time-tracking", "observer", "customer", and "external"

The "<OpUser.LinkedResource>" sub-section

If you want to automate resource creation for users then the most simple way is to add the following entries inside the <users> section:

```
<OpUser.LinkedResource synched="false">
  <OpResource.Name value="name"/>
  <OpResource.FullName value="displayName"/>
</OpUser.LinkedResource>
```

This entry automatically creates a resource for users newly synchronized from Atlassian Crowd with a resource name equal to the user name and a full name equal to the display name in Atlassian Crowd.

The following Atlassian Crowd property identifiers can be used as source for resource properties:

name	The Atlassian Crowd user name.
email	The email address.
firstName	The first name of the user.
lastName	The last name of the user.
displayName	The full (display) name of the user.
externalId	An Atlassian Crowd internal id for the user.
attribute: crowdAttributeName	A user-defined attribute with the specified name (in this case: "crowdAttributeName").

This example would prefix each resource with "Resource-" and then use the Atlassian Crowd account name as resource name. This is not recommended but only used to show a possible configuration.
The "Jira Username" of the resource will be set to the Atlassian Crowd account name and prefixed with "Jira- ". The "Jira Username" property should normally not be needed. It is only relevant if you licensed the Planforge option and the user-names in Jira differ from your desired resource names.
The resource full name is set to the display name in Atlassian Crowd.
Finally, the Planforge resource number and description will be set to the Atlassian Crowd user-defined properties resourceNumber and resourceDescription respectively.

```
     <OpUser.LinkedResource enabled="true" synched="true" autoCreate="true" assignToUserWhen="
resourceCreatedOrUnassigned">
        <OpResource.Name value="name" synched="false">
          <replace>
            <from>^(.*)$</from>
            <to>Resource-$1</to>
          </replace>
        </OpResource.Name>
        <OpResource.Number value="attribute:resourceNumber" synched="true"/>
        <OpResource.JiraUsername value="name" synched="true">
          <replace>
            <from>^(.*)$</from>
            <to>Jira-$1</to>
          </replace>
        </OpResource.JiraUsername>
        <OpResource.FullName value="displayName" synched="true"/>
        <OpResource.Description value="attribute:resourceDescription" synched="true"/>
     </OpUser.LinkedResource>
```

This specific example checks the resources every time when a user is synchronized to Planforge and assigns a resource with matching name (in this case "Resource-username") to the user if the resource is newly created or if it is not yet assigned to a user.

Element description:

<OpUser. LinkedResource enabled=" true">	Defines if resources are automatically linked/created (enabled=" true") or if the whole <OpUser.LinkedResource/> section is ignored (enabled="false").
<OpUser.LinkedResource synched="true">	Defines if the resource will only be automatically created, updated and/or assigned when creating a user in Planforge (synched="false") or each time a user logs in or is synchronized to Planforge (synched="true"). Specifying synched="false" here overrides any synched="true" attributes of the child elements of <OpUser.LinkedResource>
<OpUser. LinkedResource autoCreate=" true">	Determines if resources are automatically created (autoCreate=" true") if they do not yet exist.

<OpUser.LinkedResource assignToUserWhen=" resourceCreatedOrUnassigned" >	Handles how resources are assigned to users (the when part is influenced by "synched") Allowed values: • "never": resources are only created but never assigned to the user. This option was added for the sake of completeness only and is not recommended. • "resourceCreated": only resources which are created when the user is created are assigned. No already existing resource is touched • "resourceCreatedOrUnassigned": the resource is assigned to the user only if it is not already assigned to another user (so if the resource was created when the user logged in, or if was created earlier, but not assigned to any user) • "always": if the OpResource.Name entry produces a valid resource name, then the resource is assigned to the current user, even if it was already assigned to somebody else.
<OpResource.Name value=" name" synched="false">	The value attribute defines which Atlassian Crowd property or attribute is used for the new resource name. Sensible values are: • "name": the name (given_name + family_name) of the user. • "email": the user's email address. • "attribute:XXX": a user defined Crowd attribute
<OpResource.Number value=" attribute:resourceNumber" synched="false"/>	The resource number.
<OpResource.JiraUsername value="name" synched="true">	The Jira Username attribute of the resource. Sensible value are the same as the ones specified in "OpResource.Name". Please note that this entry is only relevant if your license contains the Jira Connector option and if the resource name in Planforge is different from the user-names in Jira. More Information about this property can be found in the Planforge's integrated documentation under Jira Connector > Resource synchronization.
<OpResource.FullName value=" displayName" synched="true">	The full name (display name) of the resource.
<OpResource.Description value="n/a" synched="false"/>	The description of the resource.
<replace>	This section can generally be omitted and should only be used when the desired resource property differs from the value provided by the Atlassian Crowd "value=" attribute or property.

\<from\>	At least one mapping source value must match this element to be replaced with the \<to\> element. Elements are evaluated in the order in which they are stated, and the first matching one is returned. Please note that this section uses Java regular expressions to determine if a term matches.
\<to\>	A string representation of the desired Planforge resource property. This value can either reference the \<from/\> match (see the OpResource.Name example) or any other arbitrary value.

Though \<OpUser.AutoCreateLinkedResource/\> can still be used to create resource mappings, it is deprecated and should be replaced with the newer \<OpUser.LinkedResource\> ... \</OpUser.LinkedResource\> configuration.

6.10.5. Group Configuration ("\<groups\>")

The section is initiated with the `<groups>` tag and closes with `</groups>` - Here's an example:

```
<groups>
    <filter mode="any">
        <parent-group>Planforge</parent-group>
        <name>Planforge</name>
    </filter>
</groups>
```

> ⚠ **Not shadowing groups**
>
> To prevent Planforge from synchronizing Crowd groups, please remove the \<groups\> section from your configuration file. Though groups will then not be created in Planforge, group-based user filtering and user-level determination will still work.

The "\<filter\>" sub-section

This sub-section inside `<groups>` defines which groups shall be synchronized to Planforge, an example:

```
<filter mode="any">
  <parent-group>Planforge</parent-group>
  <name>Planforge</name>
  <description>*Planforge*</description>
</filter>
```

> ⚠ **Managing group synchronization with ONEPOINT Projects**
>
> If this filter is omitted or contains no entries, then all Crowd groups will be synchronized to Planforge.

Filters are case-sensitive and support partial matches (via *) and single-character wild-cards (via ?). If the filtered attribute has more than one value in Crowd, then all values are checked and at least one element must match the configured filter element to satisfy the criteria. All filter sub-sections, with the exception of \<active\> can be specified multiple times.

The elements in detail:

<filter mode=" any">	The filter mode can be "all" or "any", all means that all specified filter terms must match, and "any" means that the group is synchronized as soon as one of the specified filter items matches. The above example, for instance, states that the group must be a sub-group of "Planforge", must be named "Planforge", or must mention "Planforge" somewhere in its description.
<parent-group>	A group of which the group should be a (possibly nested) member.
<name>	The group name.
<active>	If the group is active or inactive.
<description>	The group's description.

6.11. Microsoft Teams Integration

Relevant information regarding prerequisites and configurations for custom tabs in Microsoft Teams can be found at:
https://docs.microsoft.com/en-us/microsoftteams/platform/tabs/what-are-tabs

"Create a channel or group tab - Teams" - https://docs.microsoft.com/en-us/microsoftteams/platform/tabs/how-to/create-channel-group-tab

This setup guide is an extraction of the relevant information needed for Planforge.

6.11.1. Prerequisites

Please note that, according to the following quote provided by Microsoft, local hosting is not supported: "Microsoft Teams is a cloud-based product and requires that your tab content be available from the cloud using HTTPS endpoints. Teams does not allow local hosting. You must either publish your tab to a public URL or use a proxy that exposes your local port to an internet-facing URL."

This also requires that the <connect-url> entry in your configuration.oxc.xml file is your internet-facing URL (see for example chapter 6.2 "Starting Planforge automatically").

"SameSite" cookie attribute

To allow Planforge authentication when Microsoft Teams is running in a web browser, the attribute "SameSite=None;Secure;" must be added to the cookie. More information can be found at: "SameSite cookie attribute" - https://docs.microsoft.com/en-us/microsoftteams/platform/resources/samesite-cookie-update#samesite-cookie-attribute-2020-release

Please note that this requires Tomcat 9.0.28 and later or Tomcat 8.5.48 and later. For Tomcat, in the file $tomcat_home/conf/context.xml add in section <Context> (just before the </Context> tag) the line:

```
<CookieProcessor className="org.apache.tomcat.util.http.LegacyCookieProcessor" sameSiteCookies="none" />
```

In File $tomcat_home/conf/web.xml add in section <session-config> the line:

```
<secure>true</secure>
```

In the unlikely event that you also deployed other web-applications together with Planforge on the affected tomcat, please note that these settings affects all cookies for all applications.

Additional security headers

Microsoft Teams tabs are basically web-pages and thus the web server might need additional security headers for the tab to be displayed correctly. If your Web-server provides a "Content-Security-Policy" "frame-ancestors" HTTP header, please make sure that the "frame-ancestors" directive also contains the values "'self' teams.microsoft.com *.teams.microsoft.com *.skype.com"

If some of these entries are missing, then please add them to your current firewall or web-server configuration. For more information please see "Prerequisites - Teams" - https://docs.microsoft.com/en-us/microsoftteams/platform/tabs/how-to/tab-requirements

Furthermore, as the "SameSite=None;Secure" cookie attribute is required for authentication, it is considered good security practice to add the relevant "Content-Security-Policy" "frame-ancestors" header.

6.11.2. Adjusting the configuration file

As the app package for Microsoft Teams needs the Web-address of your Planforge server, this package has to be modified by the customers themselves.

1. Navigate to "https://dev.teams.microsoft.com/apps" and log in as a person who may add apps to Microsoft Teams.
2. On the left sidebar click the "Apps" icon.
3. Click the "Import App" button.
4. Select the Planforge-Teams-Plugin.zip file, which can be found inside the folder "microsoft_teams" of the downloaded Planforge zip-archive or online at https://downloads.onepoint-projects.com/tools/Planforge-Teams-Plugin.zip
5. In the side-bar navigate to "Configure/App Features".
6. Click the "Meeting extension" item.
7. Adjust the "Configuration URL" to match the <connect-url> entry in your configuration.oxc.xml file while keeping the "/teamsConfig" part.
   If your <connect-url> entry is: https://my.company.com/planforge then the "Configuration URL" should be: https://my.company.com/planforge/teamsConfig
8. Click the "Save" button to save the changes.
9. If save does not work, then please try clicking the "Group and channel app" item and try to change the URL there (the web UI seems to have a problem where saving usually fails in the "Group and channel app" tab, yet works in "Configure/App Features". However, the reverse might also be the case)
10. Make sure that you are still in "Configure/App Features".
11. Click the "Personal app" item.
12. Update the "URL" to match the <connect-url> entry in your configuration.oxc.xml file (should be exactly the same this time). Only the "Content Url" in the dialog has to be changed, the "Website Url" may stay empty.
13. Then click "Review in Teams" or "Distribute" to deploy the Planforge tab.
14. Please follow the instructions provided by the Microsoft Teams "Setup a tab" dialog.
15. After the tab is added successfully, please wait until you see the Planforge login page.
16. The Planforge tab is now set up and should work correctly.

If you are running the tab in a web browser and cannot log in please see the item "Planforge tab is displayed correctly, but you can not log in" in the "Troubleshooting" section below.

6.11.3. Troubleshooting

If Microsoft Teams is not able to add the Planforge tab, or if the tab does not work, please start troubleshooting by familiarizing yourself with the document: "Troubleshoot your Microsoft Teams app" - htt ps://docs.microsoft.com/en-us/microsoftteams/platform/resources/troubleshoot

Common Problems

1. The "Set up Planforge for a team" dialog only shows the following message: "Tunnel planforge-server.my-company.com not found":
   - The Planforge server is not running or the app setting "Configuration URL" is not correct.

2. The "Set up Planforge for a team" dialog only shows a frame which contains a broken page image and an error message like "planforge-server.my-company.com refused to connect.":
   - The Planforge server most probably does not allow the application to be opened in an iframe (a "Content-Security-Policy: frame-ancestors 'self';" or something similar is specified), see section "Prerequisites" above.

3. The "Set up Planforge for a team" dialog shows the following error message (at the bottom) "Unable to save configuration for 'Planforge' tab":
   - Please check the browser error console and look for the string "Failed to validate tab settings. Error:" (this error message might be hidden between other errors). If the error string is "Failed to validate tab settings. Error: url": Then the host part of the <connect-url>....< /connect-url> entry in "configuration.oxc.xml" is missing or not correct. If you can not find this message or you find a similar but different error then please contact Planforge Support.

4. If the Planforge tab can be set up correctly, but the page stays empty and the browser error console contains the following message: "Failed to load resource: the server responded with a status of 500 (Server Error)":
   - Then the <connect-url>....</connect-url> entry in "configuration.oxc.xml" is missing or not correct

5. If the Planforge tab is displayed correctly, but you cannot log in when Teams is running in a web browser (login works correctly in the Teams app, in the web browser the login tab screen appears again after successfully logging in):
   - The "SameSite=None;Secure" directive is usually missing from the session cookie.

7. APPENDIX

7.1. PostgreSQL Connection String

This is an example connection string for a PostgreSQL instance installed with default values on "localhost" using port 5432 and database name "planforge":

```
jdbc:postgresql://localhost:5432/planforge
```

7.2. MS SQL Server Connection String

This is an example connection string for a MS SQL instance installed with default values on "localhost" using port 1433 and database name "planforge":

```
jdbc:sqlserver://localhost:1433;database=planforge
```

> ℹ️ **Addtional parameters for MS SQL**
>
> If you need to connect to a specific instance of the MS SQL database, it can be specified by adding the following:
>
> ```
> jdbc:sqlserver://localhost;instanceName=INSTANCE;databaseName=planforge
> ```
>
> More information on additional parameters can be found at: https://docs.microsoft.com/en-us/sql/connect/jdbc/building-the-connection-url

7.3. Oracle Connection Strings

This is a connection string for a single Oracle instance installed with default values on "localhost" with SID "orcl" using port 1521:

```
jdbc:oracle:thin:@localhost:1521/orcl
```

An example for an Oracle RAC connection string:

```
jdbc:oracle:thin:@(DESCRIPTION=(LOAD_BALANCE=on)
         (ADDRESS=(PROTOCOL=TCP)(HOST=host1)(PORT=1521))
         (ADDRESS=(PROTOCOL=TCP)(HOST=host2)(PORT=1521))
         (CONNECT_DATA=(SERVICE_NAME=service)))
```