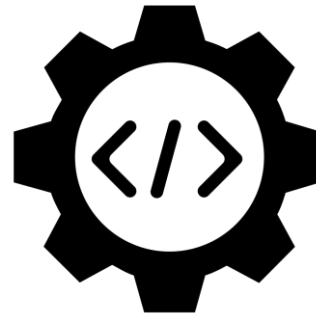


2016

SAFAX

Installation



SVN JDK Eclipse Tomcat MySQL

MongoDB GDAL





Table of Contents

Introduction	3
List of Acronyms and Definitions	4
List of Acronyms.....	4
List of Definitions	4
Prerequisites	5
SVN.....	5
Windows	5
Ubuntu	5
JDK.....	5
Windows	5
Ubuntu	6
Tomcat	7
Windows	7
Ubuntu	7
Eclipse	7
Windows and Ubuntu	7
MySQL.....	8
Windows	8
Ubuntu	8
MongoDB	9
Windows	9
Ubuntu	9
GDAL	10
Windows	10
Ubuntu	11
Development Environment.....	12
SVN.....	12
Windows	12
Ubuntu	13
Tomcat	14
Windows and Ubuntu	14



Eclipse	14
Windows and Ubuntu	14
MySQL	19
Windows and Ubuntu	19
Execution Environment.....	20
Prerequisites	20
Service Deployment.....	20
Windows and Ubuntu	20
References	22



Introduction

Cloud storage services have become increasingly popular in recent years. Users are often registered to multiple cloud storage services that suit different needs. However, the ad-hoc manner in which data sharing between users is implemented leads to issues for these users. For instance, users are required to define different access control policies for each cloud service they use and are responsible for synchronizing their policies across different cloud providers. Users do not have access to a uniform and expressive method to deal with authorization. Current authorization solutions cannot be applied *as-is*, since they cannot cope with challenges specific to cloud environments.

In order to address these challenges we have developed SAFAX [1], an extensible authorization framework offered as a service. SAFAX provides a novel XACML-based architectural framework tailored to the development of extensible authorization services for clouds. The key design principle underlying SAFAX is that all components are loosely coupled services, thus providing the flexibility, extensibility and scalability needed to manage authorizations in cloud environments. SAFAX's architecture allows users to: *a)* deploy their access control policies in a standard format; *b)* in a single location; and *c)* augment policy evaluation with information from user selectable external trust services.

The SAFAX authorization service is implemented in Java running on a Tomcat server and using Jersey service framework. MySQL is used as a back-end persistent data storage. SVN is used for version control of the SAFAX code base. Eclipse IDE is used as a development tool for modifying SAFAX code base.

This document explains how to install the SAFAX framework along with its development and execution environment.



List of Acronyms and Definitions

List of Acronyms

JDK: Java Development Kit

SAFAX: eXtensible Authorization Framework As a Service

MySQL: My Structured Query Language

SVN: Apache Subversion

XACML: eXtensible Access Control Markup Language

List of Definitions

Servlet: a server-side java program that processes and response to client requests.



Prerequisites

The following applications are prerequisites for the installation of SAFAX development and execution environment:

- SVN
- JDK
- Tomcat
- Eclipse
- MySQL
- MongoDB
- GDAL

SVN

SVN is required to download the SAFAX repository. To install SVN follow the steps below:

Windows

1. Download and install TortoiseSVN (<http://tortoisesvn.net/download.html>).

Ubuntu

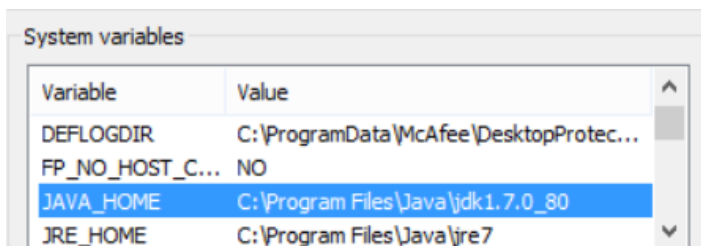
1. Use the default installed subversion in Ubuntu.

JDK

All SAFAX web services are developed in JAVA and packaged as WAR files. JDK is needed both for the development and execution environment. To install JDK follow the steps below:

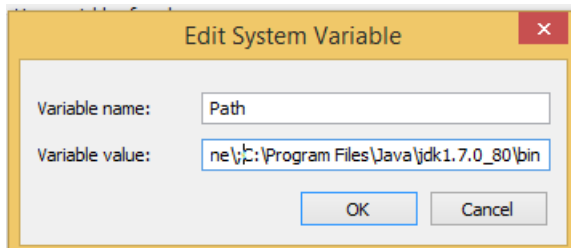
Windows

1. Download and install JDK version 7 by clicking on <http://www.oracle.com/technetwork/java/javase/downloads/index.html>.
2. Setup JAVA_HOME
 - 2.1. Right-click on *My Computer* and select *Properties*.
 - 2.2. Click on *Advanced system settings* -> *Advanced* -> *Environmental Variables*.
 - 2.3. Under *System variables*, click on *New*.
 - 2.4. In the *Variable name* enter: "JAVA_HOME".
 - 2.5. In the *Variable value* enter the path of the folder where JDK was installed (e.g. "C:\Program Files\Java\jdk1.7.0_80").



3. Setup JAVA_PATH

- 3.1. Right-click on *My Computer* and select *Properties*.
- 3.2. Click on *Advanced system settings* -> *Advanced* -> *Environmental Variables*.
- 3.3. Under *System Variables*, select variable *Path* and click on *Edit*.
- 3.4. In the *Variable value* append ";" and then the path to the Java executable (e.g. ";C:\Program Files\Java\jdk1.7.0_80\bin").

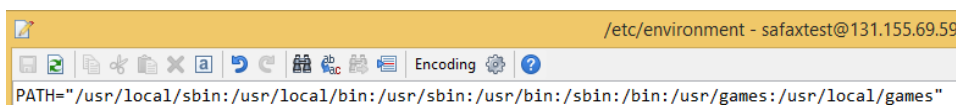


Ubuntu

1. Download and install JDK version 7
 - 1.1. Open a Terminal and type


```
$ sudo apt-get install openjdk-7-jdk
```
2. Setup JAVA_HOME
 - 2.1. Append the following line to the file */etc/environment*:


```
JAVA_HOME= <path_to_jdk_installed_directory>
```
3. Setup JAVA_PATH.
 - 3.1. In the file */etc/environment* edit the variable *Path* and append the path to the Java executable.





Tomcat

Apache Tomcat is used in this project as a servlet container for Jersey web services. It is needed for both the development and execution environment. To download and install Tomcat, follow the steps below:

Windows

1. Create a custom directory.
2. Download the 32-bit or 64-bit zip file from <https://tomcat.apache.org/download-70.cgi>
3. Extract the zip file in the custom directory.

Ubuntu

1. Navigate to this link <https://tomcat.apache.org/download-70.cgi>
2. Under *Binary Distributions* and *Core* locate the link to tar.gz.
3. In a Terminal, to download tomcat type:
`$ wget <url_link_to_tomcat_tar_gz>`
4. Decompress the file:
`$ tar xvzf apache-tomcat-7.0.53.tar.gz`
5. Move the folder into a proper location:
`$ mv apache-tomcate-7.0.53 /home/safaxtest/srv`

Tomcat install directory contains the following sub-directories

- Bin: contains the binaries and the scripts *startup* and *shutdown*.
- Conf: contains the system configuration files.
- Lib: contains the Tomcat's JAR files.
- Logs: contains the Tomcat's log files for error debugging.
- Webapps: web services are deployed in this folder.

Eclipse

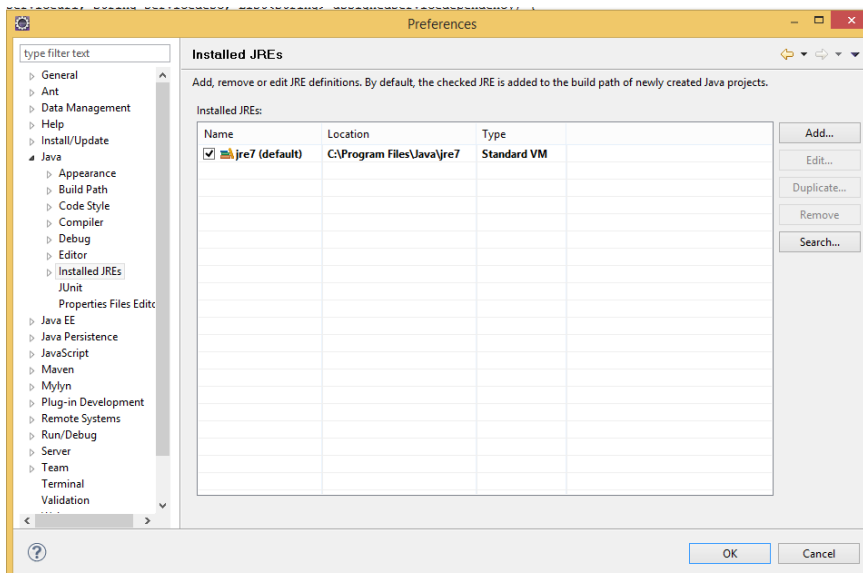
Windows and Ubuntu

Eclipse is used to modify the SAFAX framework and export SAFAX web services into .war files for their deployment on Apache Tomcat. To install Eclipse:

1. Download and install Eclipse Kepler standard (Eclipse IDE for Java EE Developers version) by clicking on <http://www.eclipse.org/downloads/packages/release/Kepler/SR2>.



2. Start Eclipse.
3. Click *Window -> Preferences -> Java -> Installed JREs*.
4. Configure Java 7 as jre.



MySQL

MySQL is used as a back-end persistent data storage for SAFAX. To install MySQL 5.5 follow the steps below:

Windows

1. Download the 32-bit or 64-bit Windows MSI installer from <https://dev.mysql.com/downloads/mysql/5.5.html>.
2. Double click on the executable installation file, and follow the instructions

Ubuntu

1. Open a Terminal and type:



```
$ sudo apt-get update
```

```
$ sudo apt-get install mysql-server-5.5
```

2. To start MySQL, type:

```
$ sudo service mysql start
```

3. Check if MySQL is running by running:

```
$ sudo netstat -tap | grep mysql
```

MongoDB

MongoDB is used as a back-end persistent data storage for SAFAX log. To install MongoDB 3.2 follow the steps below:

Windows

3. Download the 32-bit or 64-bit Windows MSI installer from

<https://www.mongodb.com/download-center>

4. Double click on the executable installation file, and follow the instructions

Ubuntu

1. Open a Terminal and type:

```
$ sudo apt-key adv --keyserver hkp://keyserver.ubuntu.com:80 --recv EA312927
```

```
$ echo "deb http://repo.mongodb.org/apt/ubuntu precise/mongodb-org/3.2 multiverse" | sudo tee /etc/apt/sources.list.d/mongodb-org-3.2.list (Ubuntu 12.04)
```

```
$ sudo apt-get update
```

```
$ sudo apt-get install -y mongodb-org=3.2.8 mongodb-org-server=3.2.8 mongodb-org-shell=3.2.8 mongodb-org-mongos=3.2.8 mongodb-org-tools=3.2.8
```

2. To start MongoDB, type:

```
$ sudo service mongod start
```

3. Check if MySQL is running by checking the contents of the log file at:

```
/var/log/mongodb/mongod.log for a line reading
```

```
[initandlisten] waiting for connections on port <port>
```



GDAL

GDAL is a translator library for raster and vector geospatial data formats. It is used in SAFAX because it provides the following methods:

- `CreateFromGML(String gml)`: create geometry from GML
- `CreateFromWkt(String wkt)`: create geometry from well known text format
- `ExportToGML()`: convert a geometry into GML format
- `ExportToWkt()`: convert a geometry into well known text format

Windows

1. Download the 32-bit or 64-bit Windows installer from <http://www.gisinternals.com/release.php>
2. Unzip the rar file to a folder.
3. Right click on *Computer* and choose *Properties* and then choose *Advanced system settings*.
4. Add the following paths to the Environments variables:
 - 4.1. `GDAL_DATA` with value `\your_unzip_folder\bin\gdal-data`
 - 4.2. `GDAL_DRIVER_PATH` with value `\your_unzip_folder\bin\gdal\plugins`
 - 4.3. `PROJ_LIB` with values `\your_unzip_folder\bin\gdal\projlib`
5. Finally, add the following paths to path variable:
 - 5.1. `\your_unzip_folder\bin\gdal\java`
 - 5.2. `\your_unzip_folder\bin\gdal\java`
6. Copy the following files from `\your_unzip_folder\bin\gdal\java\` folder to apache tomcat lib folder:
 - 6.1. `gdal.jar`
 - 6.2. `gdalconstjni.dll`
 - 6.3. `gdaljni.dll`
 - 6.4. `ogrjni.dll`
 - 6.5. `osrjni.dll`
7. Restart the computer.



Ubuntu

1. Open a Terminal and type:

```
$ sudo apt-key install build-essential ant swig
```

```
$ sudo apt-get install libproj0
```

```
$ svn checkout https://svn.osgeo.org/gdal/trunk gdal-trunk
```

```
$ cd gdal-trunk/gdal
```

```
$ ./configure
```

```
$ make
```

```
$ sudo make install
```

```
$ cd /swig/java
```

(Modify java.opt with appropriate java local version: JAVA_HOME, JAVADOC, JAVAC, JAVA, JAR, JAVA_INCLUDE)

```
$ make
```

```
$ ls | grep .so
```

```
$ cp *.so *.jar *.la /usr/local/lib
```

```
$ export LD_LIBRARY_PATH=/usr/local/lib
```

```
$ sudo ldconfig
```

2. Move to tomcat bin folder and modify the setenv.sh file. Add the following line to the file:
export JAVA_OPTS="-Djava.library.path=/usr/local/lib".
3. Copy .so gdal.jar .la files from /usr/local/lib to tomcat lib folder



Development Environment

The following applications are needed to install the SAFAX development environment:

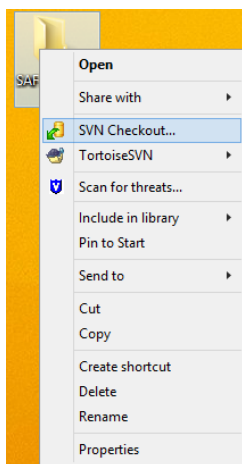
- SVN
- Tomcat
- Eclipse
- MySQL
- MongoDB
- GDAL

See Section “Prerequisite” for the installation of these applications.

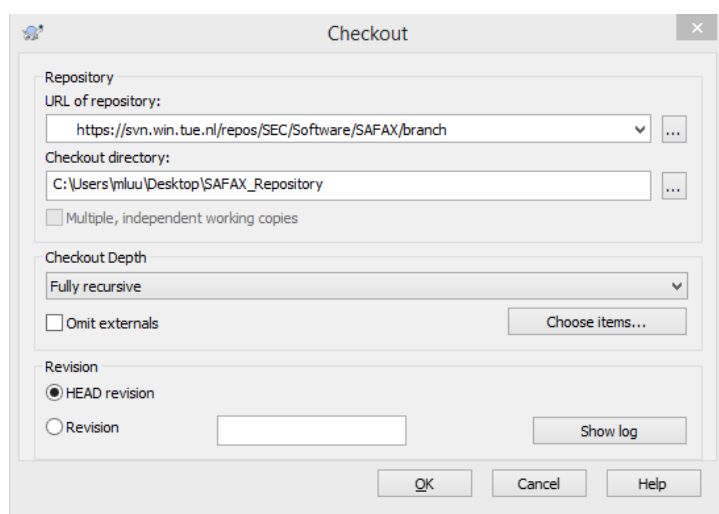
SVN

Windows

1. Create a new folder (e.g. SAFAXRepository), right-click on the folder, and click on *SVN Checkout*.



2. Enter SAFAX SVN account provided by SAFAX administrators. The URL of repository is: <https://svn.win.tue.nl/repos/SEC/Software/SAFAX/branch>.





Ubuntu

1. Open a Terminal and type:

```
$ svn co --username yourname  
https://svn.win.tue.nl/repos/SEC/Software/SAFAX/branch/PROJECTNAME
```

The following folders can be found in SAFAX repository:

- *ch* contains the context handler web service.
- *credential* contains the credential service.
- *herasaf* contains the PDP implementation provided by HERAS-AF.
- *pap* contains the PAP service providing interfacing for users to deploy their access control policies.
- *pep* contains the PEP service providing interfacing for consumers to send an access request to SAFAX for evaluation.
- *pip* contains the PIP service. The PIP service is used by context handler to access and manipulate the data contained in the data sources.
- *reputation-eb* contains the Evidence-based reputation service. It is an external trust service that computes the reputation of a user based on reputation policies that are represented in Prolog format.
- *reputation-fb* contains the Flow-based reputation service. It is an external trust service that computes the reputation of a user based on reputation policies that are represented in Prolog format.
- *sfx* is SAFAX UI. It consists of HTML, CSS, and JavaScript files. It is primarily designed as a frontend and interacts with the SFX backend services to populate and provide data to SAFAX framework.
- *sfxservice* contains SAFAX backend services such as logging users, establishing guest sessions, creating projects, demos among many other features.
- *similarity* contains the similarity service. Within SAFAX, this service is primarily used for policy alignment where policy and the access request might use different terms but with the same semantic meaning.
- *sr* contains the service registry service. It is a central component of the SAFAX framework allowing various services to discover each other and also contains meta information related to each service such as input parameter type, input parameter number, output format, messaging format, nature of service (PDP, PAP, UDF, etc.,).



Tomcat

Windows and Ubuntu

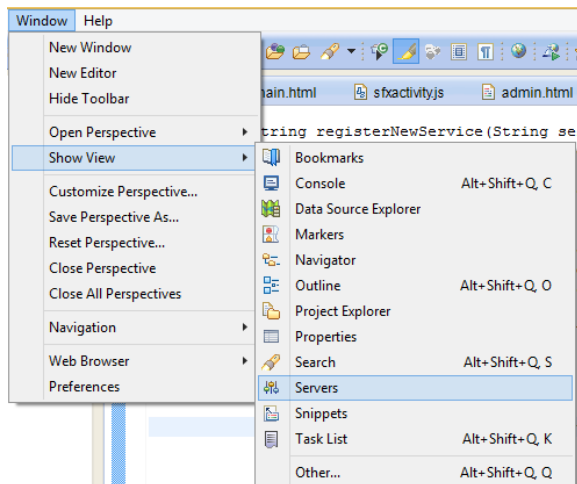
1. Tomcat installed directory contains the following sub-directories:
 - 1.1. Bin: contains the binaries and startup script and shutdown script.
 - 1.2. Conf: contains the system configuration files.
 - 1.3. Lib: contains the Tomcat's JAR files.
 - 1.4. Logs: contains the Tomcat's log files for error debugging.
 - 1.5. Webapps: SAFAX web services are deployed in this folder.
2. Change tomcat port to 80 (tomcat-installed-folder/conf/server.xml)

```
<Connector port="80" protocol="HTTP/1.1"
connectionTimeout="20000" redirectPort="8443" />
```

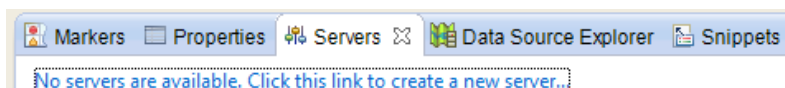
Eclipse

Windows and Ubuntu

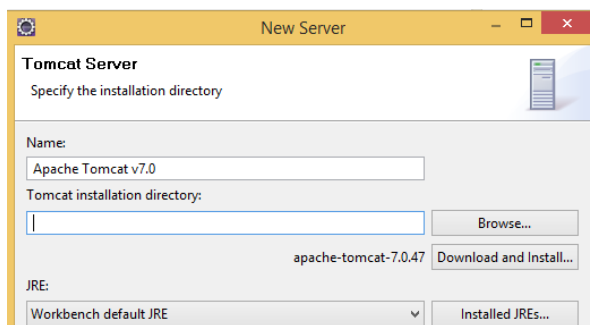
1. Run Eclipse.
2. Go to the *Window -> Show View -> Servers*



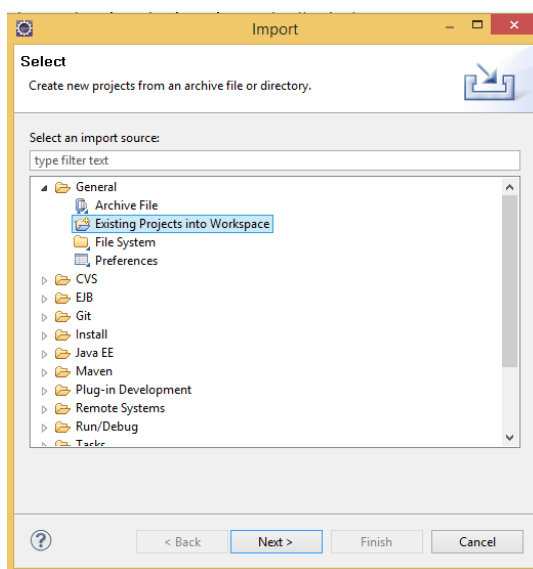
3. Go to the *Servers* tab at the bottom of Eclipse and click the link to add new server.



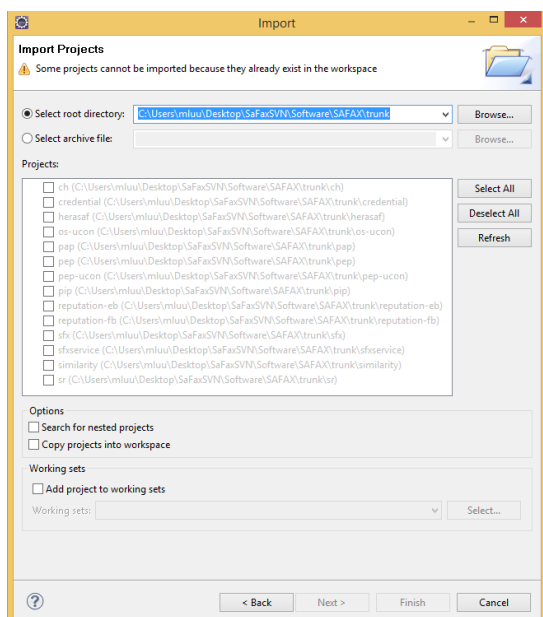
4. Choose Tomcat v7.0 Server in the list and click *Next*.
5. Indicate the directory of Tomcat 7 in the computer.



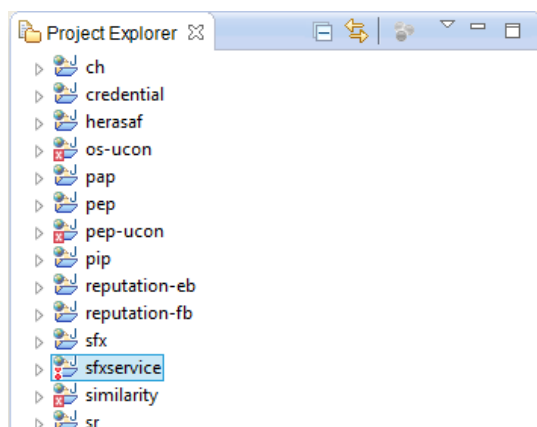
6. Double click on the created server and set the port to 80.
7. Click *File -> Import -> Existing Projects into Workspace*. Then click *Next*.



8. In *Select root directory*, click browse and choose SAFAX trunk or tags or branches folder.
9. Select all the projects listed in the root folder and click Finish.

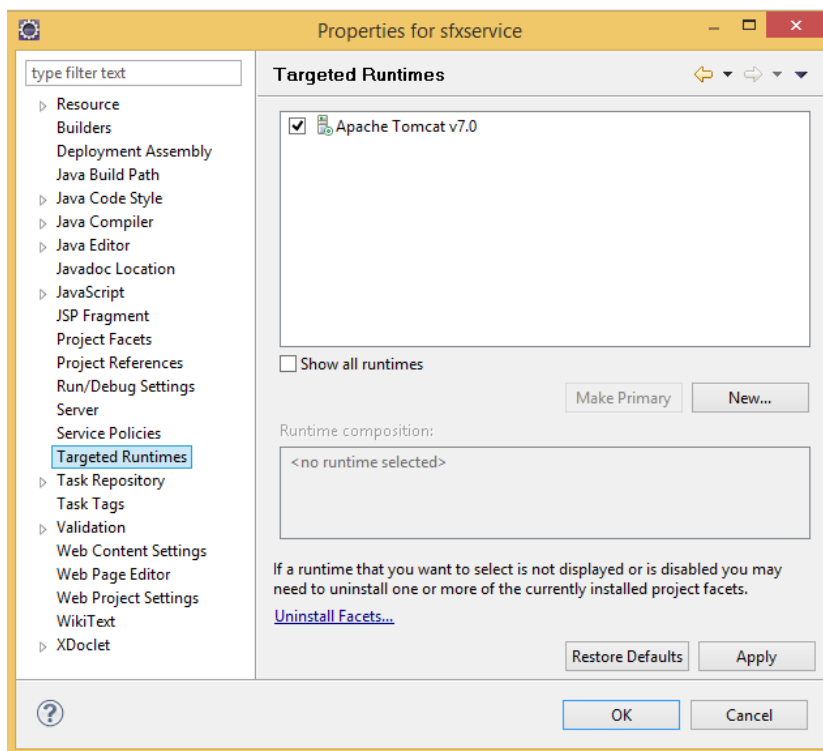


10. Now a list of SAFAX projects is displayed in Eclipse work space.



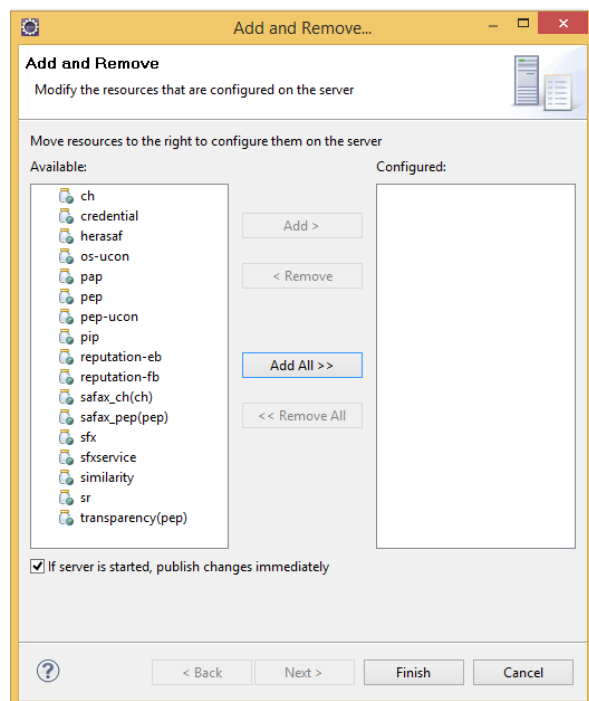
11. Right-click on sfxservice project and click *Properties* -> *Targeted Runtimes*.

12. Choose *Apache Tomcat v7.0* under *Targeted Runtimes* tab.

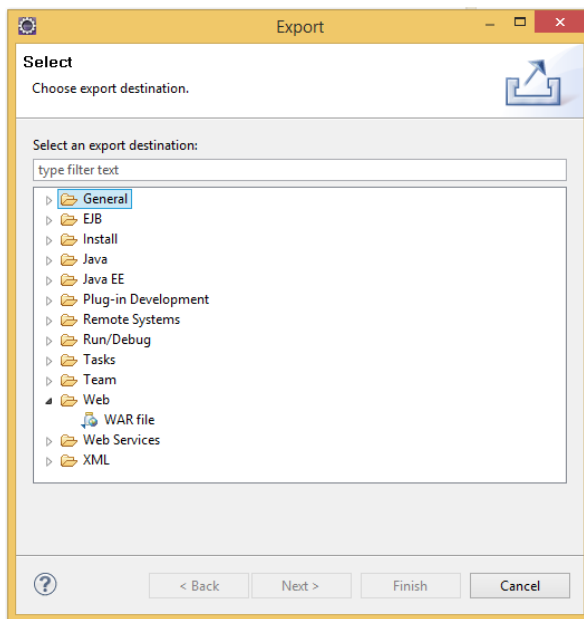


13. Right-click on Tomcat Server icon under *Servers* tab. Click *Add and Remove*.

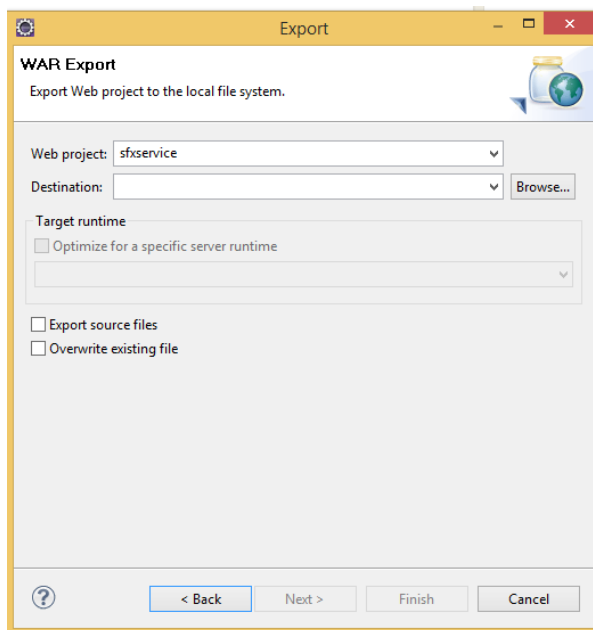
14. In the *Add and Remove* pop up window, click *Add All*.



15. Click *File* -> *Export* to export SAFAX web projects into .war files.



16. In the *Web project* section, choose the SAFAX project that will be exported.



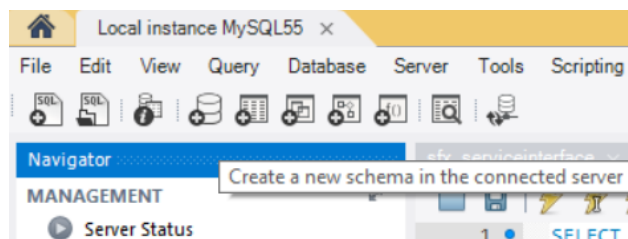
17. In the *Destination* section, choose the location where .war files are stored. Click *Finish*.



MySQL

Windows and Ubuntu

1. Use MySQL Workbench to create a new database and importing SAFAX database schema.
2. Click *Create a new schema in the connected server* button



3. Name the schema: safax. Choose *latin1_swedish_ci* for the Collation.

Collation:

4. Click *Open SQL Script* to import the SAFAX initial database. The dump script of the initial database can be found in the SAFAX repository (<https://svn.win.tue.nl/repos/SEC/Software/SAFAX/sql>).
5. Execute the script to create tables in the SAFAX database.



Execution Environment

Prerequisites

The following applications are needed to install the SAFAX development environment:

- JDK
- Tomcat
- MySQL

See Section “Prerequisite” for the installation of these applications.

Service Deployment

Windows and Ubuntu

1. Change tomcat users (tomcat-installed-folder/conf/tomcat-users.xml)

```
<role rolename="tomcat"/>
<role rolename="manager-gui"/>
<role rolename="manager-script"/>
<role rolename="manager-jmx"/>
<role rolename="manager-status"/>
<role rolename="admin-gui"/>
<role rolename="admin-script"/>
<user username="tomcat" password="tomcat" roles="tomcat"/>
<user username="both" password="tomcat" roles="tomcat,role1"/>
<user username="role1" password="tomcat" roles="role1"/>
<user username="admin" password="admin" roles="manager-gui,manager-script,manager-
jmx,manager-status,admin-gui,admin-script"/>
```

2. Change tomcat port to 80 (tomcat-installed-folder/conf/server.xml)

```
<Connector port="80" protocol="HTTP/1.1"
connectionTimeout="20000" redirectPort="8443" />
```

3. Deploy SAFAX web services to Tomcat server by using one of the two options:

- 3.1. Copying web application archive file (.war) into Tomcat webapps folder. .war files can be found in SAFAX SVN server.

- 3.2. Using Tomcat’s manager application:

- 3.2.1. Navigate to Tomcat server URL.

- 3.2.2. Go to Manager App (top right).

- 3.2.3. Login by using the username and password mentioned in tomcat-users.xml file.

- 3.2.3.1. Scroll down to the section ‘Deploy’.



3.2.3.2. Upload the .war file and click 'Deploy'.

4. Stop/Start Tomcat

4.1. Open terminal

4.2. Type `cd ~/tomcat-installed-directory`

4.3. Type `sudo ./bin/shutdown.sh` to stop

4.4. Type `sudo ./bin/startup.sh` to start.



References

- [1] S. P. Kaluvuri and A. I. Egner and J. den Hartog and N. Zannone. SAFAX -- an extensible authorization service for cloud environments. *Frontiers in ICT* 2(9), 2015.
- [2] Apache Tomcat 7 Installation [Internet]. [cited 2015 Oct 26]. Available from: <https://tomcat.apache.org/tomcat-7.0-doc/appdev/installation.html>