Submitted to the EC on XXX/XX/20XX

COMPETITIVENESS AND INNOVATION FRAMEWORK PROGRAMME

ICT Policy Support Programme (ICT PSP)

e-CODEX

e-Justice Communication via Online Data Exchange

ICT PSP call identifier: CIP-ICT-PSP-2009-4

ICT PSP main Theme identifier: CIP ICT PSP 2010 5.2 3: E-JUSTICE SERVICES

Project full title: e-Justice Communication via Online Data Exchange

Grant agreement n°: 270968

Installation Description

Deliverable ID:	XX
Deliverable Name :	Installation Description
Status :draft	V0.9
Dissemination Level :	e-CODEX consortium
Due date of deliverable :	XX
Actual submission date :	[Pending]
Work Package :	e-CODEX WP5
Organisation name of lead partner for this deliverable :	Bundesministerium für Justiz, Ministerio de la Justicia
Author(s):	AT, ES, DE
Partner(s) contributing :	DE, EE,IT

The main goal of this document is to guide the administrators through the installation of the e-Codex Gateway.

History

Version	*Date*	*Changes made*	*Modified by*
0.1	09.10.2012	First draft	AT
0.9	18.10.2012	First version	AT
1.0	29.10.2012	Second version	SP
1.3	26.09.2013	Complete rework for release of version 1.3	DE
1.4	12.02.2014	Updated Info for Release 1.4	DE

Table of contents

HISTORY **2**

1. GLOSSARY **4**

2. SCOPE OF THIS DOCUMENT **5**

3. OVERVIEW **6**

4. MAKE SURE YOUR SYSTEM MEETS THE INSTALLATION REQUIREMENTS **7**

4.1. System Overview 7

4.2. Services 7

4.3. Software Requirements 7

4.4. Database Requirements 7

5. OBTAINING THE SOFTWARE **8**

5.1. Software Packages 8

5.1.1. holodeck-distribution-${version}.war 8

5.1.2. holodeck-distribution-${version}-distribution-war.zip 8

5.1.3. holodeck-distribution-${version}-tomcat-full.zip 8

5.1.4. holodeck-distribution-${version}-tomcat-configuration.zip 9

5.1.5. holodeck-distribution-${version}-sql-scripts.zip 9

5.2. Downloading the software 9

5.3. Download the PMode and security configuration files 10

6. CONFIGURE YOUR PROXY SERVER **12**

6.1. SSL configuration 12

7. CONFIGURE YOUR DATABASE **13**

8. INSTALL THE GATEWAY SOFTWARE **14**

8.1. Deploy the database driver 14

9. CONFIGURE THE GATEWAY SOFTWARE **15**

9.1. context.xml 15

9.2. hibernate.properties 17

9.3. PModes and public certificates 17

9.4. Private key 18

9.5. security-config.xml 18

</TNS:KEYSTORES> **18**

10. START THE SERVER **19**

11. FURTHER RESOURCES **21**

11.1. Level of support 21

11.2. e-CODEX JIRA Server 21

1. Glossary

e-CODEX gateway	GW

2. Scope of this document

This document is intended for server administrators who are tasked with installing the e-CODEX gateway software. Basic knowledge of operating systems, proxy servers, servlet containers and networking is assumed. This manual is neither intended to be a guide to e-CODEX in general, nor does it contain the preliminary steps required to participate in the e-CODEX project. If you are looking for this kind of information please have a look at the quick start guide which has yet to be written.

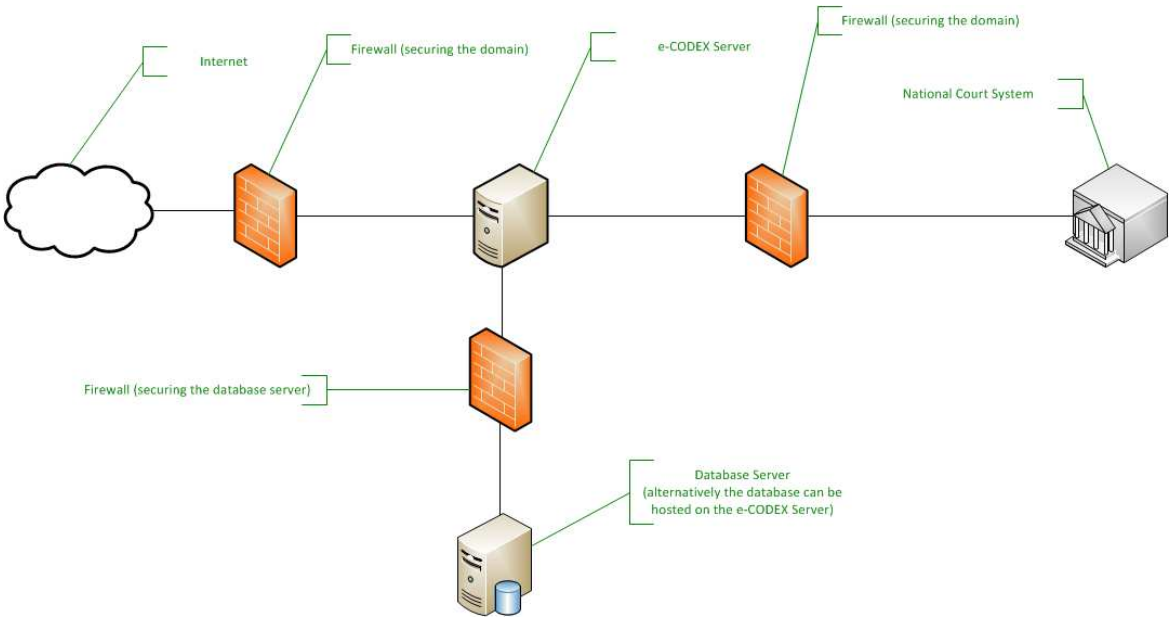
3. Overview

This manual is a step-by-step guide to installing the e-CODEX gateway v1.3 from scratch. If you are performing an update instead please have a look at the upgrade_info.txt accompanying your software download. Here is the list of steps to perform:

1. Make sure your system meets the installation requirements.
2. Download the software
3. Download the PMode and security configuration files
4. Configure your proxy server
5. Set up the database
6. Install the gateway software
7. Configure the gateway software
8. Start the server

4. Make sure your system meets the installation requirements

4.1. System Overview



The e-CODEX gateway [GW] must be installed on a server in a secure networking area behind a firewall and a proxy server (not shown). The database server used by the application can either be deployed to the same server as the gateway or a special database server.

4.2. Services

The GW exposes a number of services, most of which may only be exposed to internal users. To ensure that only the required services are reachable from the outside, the use of a proxy server (i.e. Apache2) is highly recommended. For the configuration of the proxy server please see chapter 6

4.3. Software Requirements

A Java runtime environment Version 6 or 7 is required for the deployment of the software. The use of version 7 is highly recommended as version 6 is not supported any more. For downloads and support please see http://java.com.

As the GW is a Java Servlet application a servlet container is required. The recommended and supported servlet container is Apache Tomcat (http://tomcat.apache.org/) versions 6 and 7. A GW version bundled with a Tomcat 6 container can be downloaded from the projects repository (see 5.2).

4.4. Database Requirements

Supported databases for the GW are MySQL5 and Oracle 10 + 11. While initial database scripts can be provided for all databases supported by Hibernate 3.4.0 on request, update scripts (in case of changes to the database model) will only be provided for the supported database engines. While the download packages contain database scripts for Apache Derby use of this database engine is only recommended for testing purposes and NOT for productive use.

5. Obtaining the Software

5.1. Software Packages

There are different release packagings available for the e-CODEX gateway:

Filename: holodeck-distribution-${version}	Gateway WAR-file	Fully configured Tomcat 6 Server	Tomcat server configuration	SQL-scripts for initial database generation (MySQL5, Oracle, Derby) and update scripts (MySQL5, Oracle)
.war	X			
-distribution-war.zip	X		X	X
-tomcat-full.zip	X	X	X	X
-tomcat-configuration.zip			X	
-sql-scripts.zip				X

5.1.1. holodeck-distribution-${version}.war

This package contains only the web application. The usage of this package is only recommended for development and testing purposes

5.1.2. holodeck-distribution-${version}-distribution-war.zip

This package contains the web application, tomcat configuration files and database scripts. Use this package if you are performing an update of an existing installation of version 1.3-beta-1 or higher or if you want to deploy the software to a tomcat server different from the one that is available with the GW software.

5.1.3. holodeck-distribution-${version}-tomcat-full.zip

This package contains a Tomcat 6 server with a preinstalled GW. This package is the easiest way to deploy the GW into a production environment. Usage of this package is recommended, and this manual assumes this package is used for deployment. The structure of the package is shown below:

```
|   changelog.txt
|   upgrade-info.txt
|
+---holodeck
    |
    +---common
    |   +---lib
    |
    +---conf
    |   |   catalina.policy
    |   |   catalina.properties
    |   |   context.xml
    |   |   logging.properties
    |   |   server.xml
    |   |   tomcat-users.xml
    |   |   web.xml
```

```
|   |
|   +---Catalina
|   |   +---localhost
|   |           holodeck.xml
|   |
|   +---holodeck
|       |   gateway.xml
|       |   hibernate.properties
|       |   workers.xml
|       |
|       +---keys
|       |
|       +---pmodes
|       |       pmodes.xsd
|       |
|       +---policies
|               policy-sign-body-header.xml
|               policy-sign_body_header-encrypt_body.xml
|
+---lib
+---logs
+---sql-scripts
|       holodeck_derby_initial.sql
|       holodeck_mysql_initial.sql
|       holodeck_oracle_initial.sql
|
+---store
|   +---backend_store
|   +---receive
|
+---webapps
    |   holodeck.war
```

5.1.4. holodeck-distribution-${version}-tomcat-configuration.zip

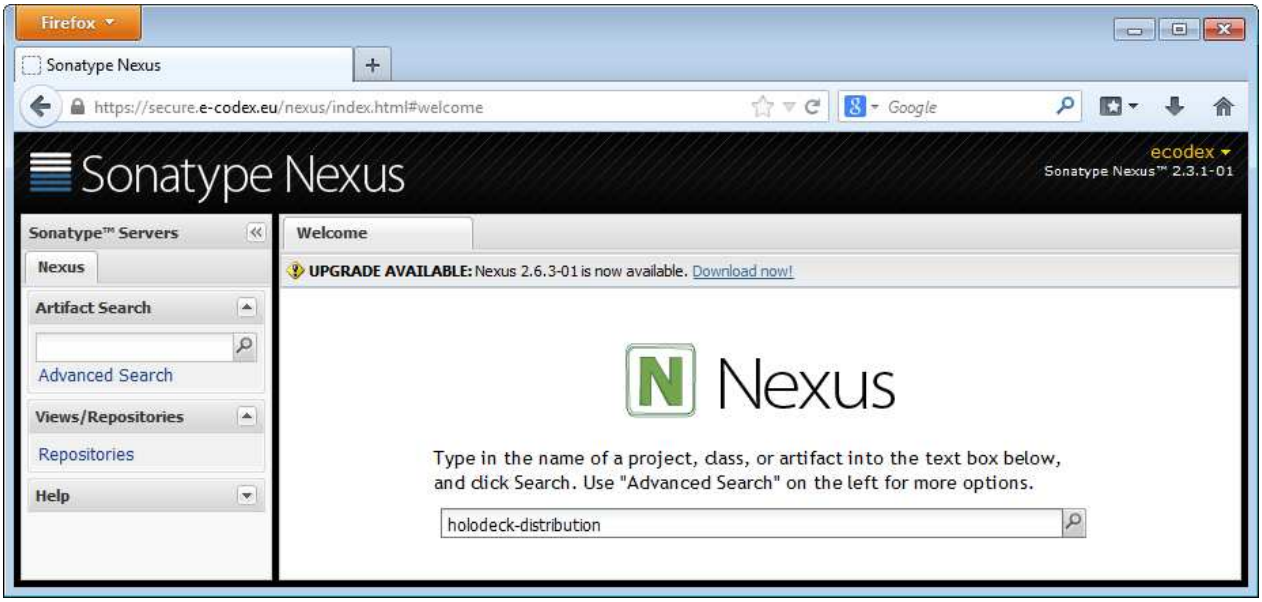
This package contains only the configuration files for the Tomcat server.

5.1.5. holodeck-distribution-${version}-sql-scripts.zip

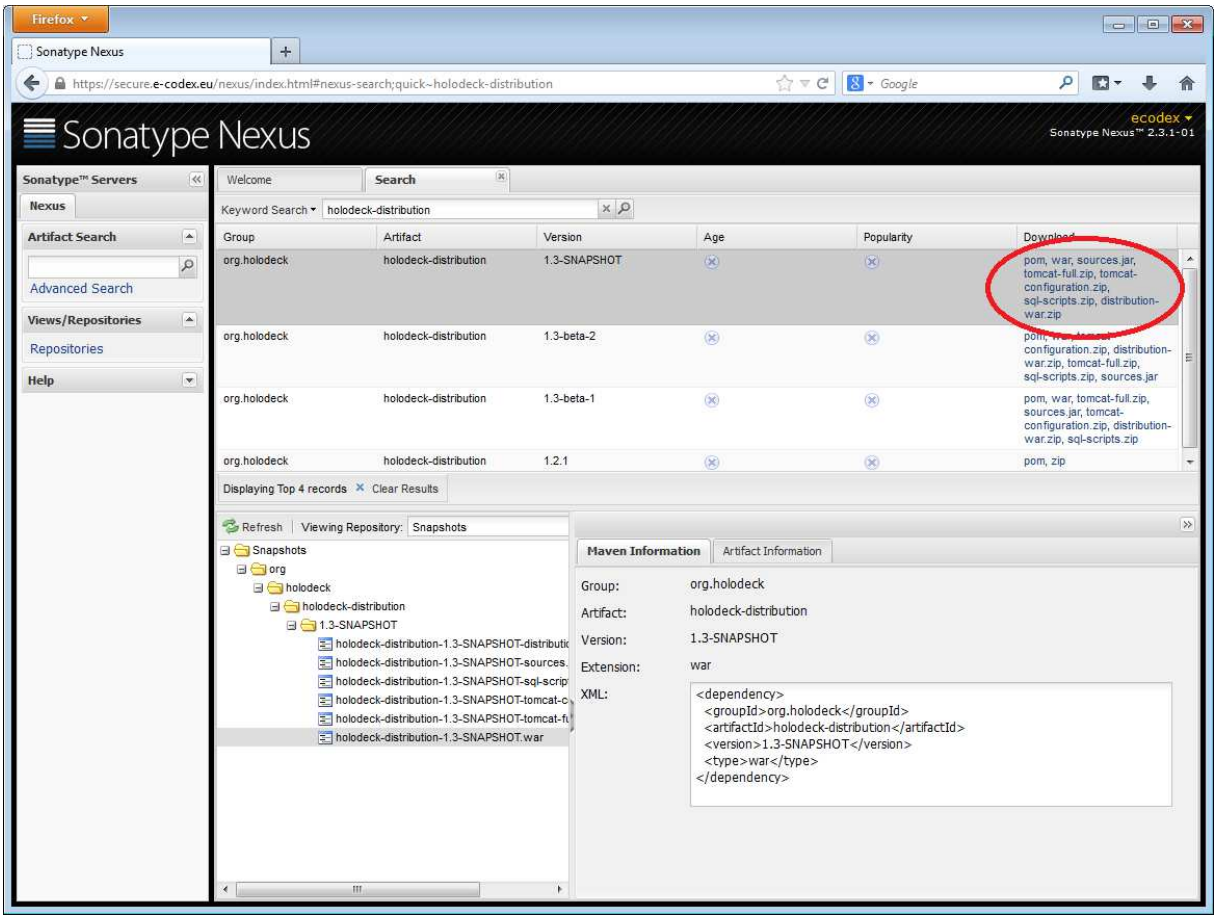
This package contains only initial and delta SQL scripts for MySQL and Oracle database engines.

5.2. Downloading the software

All e-CODEX software can be downloaded from the projects repository located at https://secure.e-codex.eu/nexus. Required credentials will be provided to you by the local person responsible for e-CODEX. After logging in, a search for holodeck-distribution gives you access to the software.



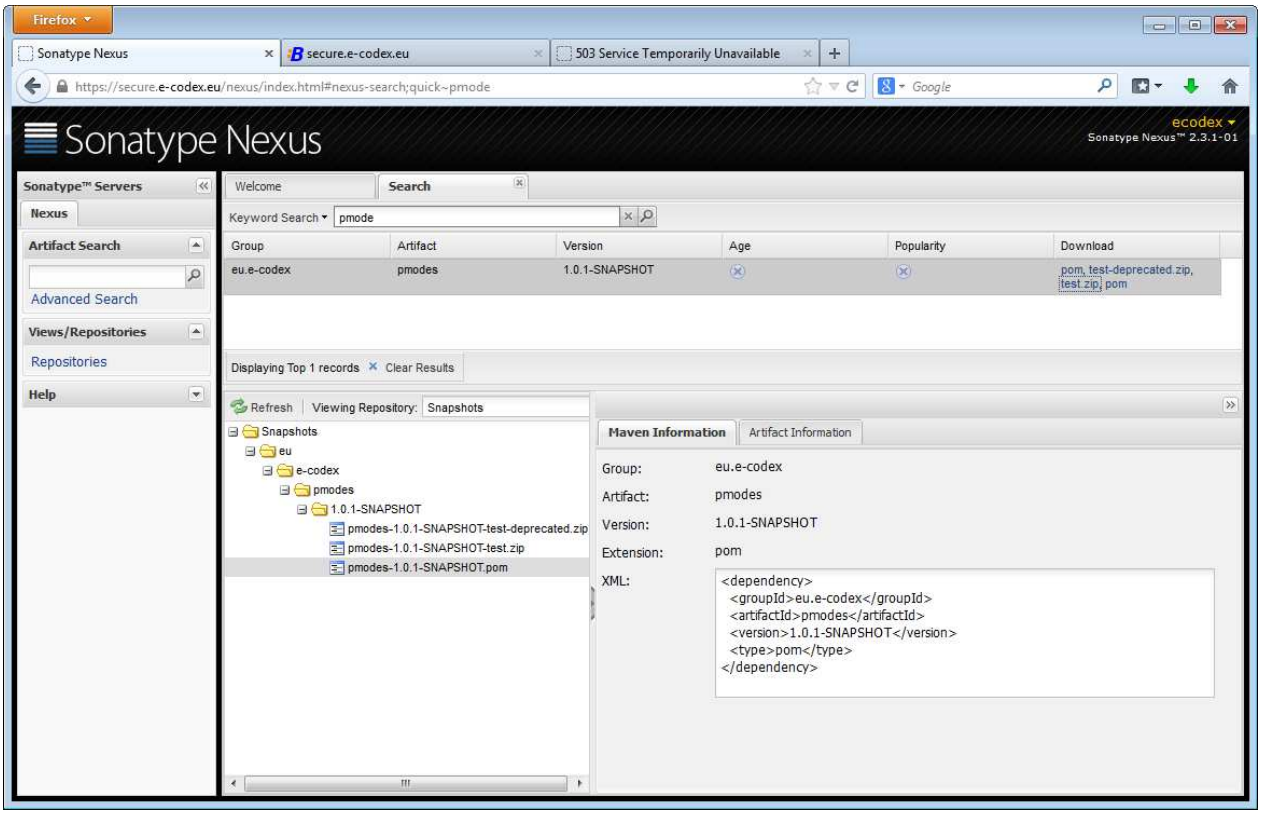
The different packages can be chosen in the “download” column at the right side. Please make sure you download the correct version.



5.3. Download the PMode and security configuration files

Processing modes (PModes) define the parameters (Endpoint addresses, security configurations etc.) for the communication between different GWs. As those properties are subject to change those

configuration files are distributed separately from the GW. There are two sets of files: testing and production. Those files are also hosted on the projects nexus server.



The –deprecated package is for the use with the GW version 1.2.1. If you are using version 1.3 or above use the package without this qualifier. To download production PModes you need special credentials. If you do not have access to those PModes please contact Tim-Marco.Nowosadtko@it.nrw.de.

6. Configure your proxy server

The GW exposes a number of different services listed in the table below. Please configure your proxy server in a way that only authorized clients can reach the services intended for them (i.e. if you are using an Apache2 webserver as proxy use ProxyPass and Allow directives for the configuration, see http://httpd.apache.org/docs/2.2/mod/mod_proxy.html). The standard server distributed in the tomcat-full package listens on http://localhost:8080 and the context path for the application is /holodeck. IF YOU DO NOT SECURE THE ACCESS TO THE ADMINISTRATIVE SERVICES THE CONFIDENTIALITY AND INTEGRITY OF YOUR DATA IS AT RISK!

Service localhost:8080/holodeck	Allow ONLY connection from	description
/	Server administrators	Axis2 status page
/services/listServices	Server administrators	Overview of deployed axis2 webservices
/axis2-admin/	Server administrators	Axis2 administration console, user:admin password:axis
/services/BackendService	National connector	Webservice for communicating with the national systems
/services/msh	e-CODEX gateways	Webservice for communication with other e-CODEX gateways

6.1. SSL configuration

The use of SSL for the communication with other GWs is mandatory. The standard configuration assumes that the SSL connection terminates at the proxy server. If you are unsure of the required configurations, please check the documentation of your proxy server (in case of Apache2 http://httpd.apache.org/docs/2.2/ssl).

7. Configure your database

If you are performing an initial deployment of the GW software you need to create a database user and a matching database. Instructions how to do this can be found in the manual of your database software. After that execute the database scripts appropriate for your database engine. Those scripts can be found in the sql-scripts folder of your software package, named holodeck_${database_engine}_initial.sql. If you are performing a software update instead of a full install please refer to the upgrade_info.txt on which scripts you have to execute.

8. Install the gateway software

To install the GW full distribution just copy the unzipped /holodeck folder to your server. If you are using an existing servlet container deploy the holodeck.war file according to your containers documentation. Additionally you have to copy the configuration files described in chapter 9 to their respective locations.

8.1. Deploy the database driver

Because of licensing reasons the distributed Tomcat server does not contain any database drivers besides an Apache Derby driver for testing purposes. A JDBC compatible database driver has to be deployed to the server (holodeck/common/lib).

9. Configure the gateway software

The following configuration files have to be modified according to your local environment:

9.1. holodeck.xml

The file conf/Catalina/localhost/holodeck/holodeck.xml contains most of the configuration data required to run the GW. During runtime those parameters are bound to the JNDI of the servlet container under the path java:comp/env/. If you use a servlet container different than Tomcat make sure those values are provided by the server. Details on how to accomplish this can be found in the documentation of your servlet container.

```
<Context path="/holodeck" privileged="false" reloadable="false">

    <!-- DATABASE CONFIGURATION START -->
    <!-- Database configuration, driverClassName, url, username and password
        must be changed -->
    <Resource name="holodeckPU"
              auth="Container"
              factory="org.apache.naming.factory.BeanFactory"
              type="com.mchange.v2.c3p0.ComboPooledDataSource"
              driverClass="org.apache.derby.jdbc.ClientDriver"
              maxPoolSize="50"
              minPoolSize="15"
              acquireIncrement="3"
              acquireRetryAttempts="0"
              acquireRetryDelay="3000"
              breakAfterAcquireFailure="false"
              maxConnectionAge="60"
              maxIdleTime="30"
              maxIdleTimeExcessConnections="10"
              idleConnectionTestPeriod="15"
              testConnectionOnCheckout="false"
              preferredTestQuery="SELECT 1"
              debugUnreturnedConnectionStackTraces="true"
              autoCommitOnClose="true"
              jdbcUrl="jdbc:derby://localhost:1527/holodeck;create=true"
              user="root"
              password="root"/>

    <!-- Name of the database connection as defined above -->
    <Environment name="holodeck.persistence.unit" value="holodeckPU"
                 type="java.lang.String"/>

    <!-- File where addidtional persistence properties are defined -->
    <Environment name="holodeck.persistence.properties"
value="conf/holodeck/hibernate.properties"
                 type="java.lang.String"/>
    <!-- DATABASE CONFIGURATION END -->


    <!-- SECURITY CONFIGURATION START -->
```

```
    <!-- File where the security module is configured -->
    <Environment name="holodeck.module.security.configFile"
                 value="conf/holodeck/security-config.xml"
type="java.lang.String"/>

    <!-- Folder where security policies are defined -->
    <Environment name="holodeck.module.security.policiesFolder"
                 value="conf/holodeck/policies" type="java.lang.String"/>

    <!-- Timeout in milliseconds for the expiration of the security configuration
        cache -->
    <Environment name="holodeck.module.security.config.timeout"
                 value="30000" type="java.lang.String"/>

    <!-- Password for the private key used by the gateway -->
    <Environment name="holodeck.module.security.config.privatekey.password"
                 value="apache" type="java.lang.String"/>
    <!-- SECURITY CONFIGURATION END -->


    <!-- FILE SYSTEM CONFIGURATION START -->
    <!-- Folder where the backend module stores messages -->
    <Environment name="holodeck.module.backend.messagesFolder"
                 value="store/backend_store" type="java.lang.String"/>

    <!-- Folder where the ebms3 module stores attachments -->
    <Environment name="holodeck.module.ebms3.attachmentFolder"
                 value="store/attachments" type="java.lang.String"/>

    <!-- Folder where the PMode configuration files are located -->
    <Environment name="holodeck.module.ebms3.PModesDir"
value="conf/holodeck/pmodes"
                 type="java.lang.String"/>

    <!-- File where gateway consumption rules are defined -->
    <Environment name="holodeck.module.ebms3.gatewayConfigFile"
                 value="conf/holodeck/gateway.xml" type="java.lang.String"/>

    <!-- Folder where the ebms3 module stores submitted messages -->
    <Environment name="holodeck.module.ebms3.submittedMessagesFolder"
                 value="store/send" type="java.lang.String"/>

    <!-- Folder where the ebms3 module stores received messages -->
    <Environment name="holodeck.module.ebms3.receivedMessagesFolder"
                 value="store/receive" type="java.lang.String"/>

    <!-- File where worker execution rules are defined -->
    <Environment name="holodeck.module.ebms3.workersFile"
                 value="conf/holodeck/workers.xml" type="java.lang.String"/>

    <!--  support for holodeck 1.3 messages -->
    <Environment name="holodeck.module.ebms3.enforce.1_3.compatibility"
                 value="true" type="java.lang.Boolean"/>
    <!-- FILE SYSTEM CONFIGURATION END -->


    <!-- MISC CONFIGURATION START -->
    <!-- Number of days after which not downloaded messages are deleted -->
    <Environment name="holodeck.module.backend.messagesTimeLive"
```

```
                value="60" type="java.lang.String"/>

    <!-- Cron expression determining how often the check for the time limit
        defined above is executed -->
    <Environment name="holodeck.module.backend.deleteMessagesCron"
                value="0 1/2 * * * ?" type="java.lang.String"/>

    <!-- Hostnames of the server the gateway is running on -->
    <Environment name="holodeck.module.ebms3.hostnames"
value="localhost,127.0.0.1"
                type="java.lang.String"/>
    <!-- MISC CONFIGURATION END -->
</Context>
```

If you are using the full distribution, editing the "DATABASE CONFIGURATION" and "SECURITY CONFIGURATION" sections of this file is sufficient. Otherwise you will also have to edit the "FILE SYSTEM" section and create the respective folders and copy the defined configuration files to the specified locations. Relative paths in this configuration file are always relative to the folder your server is started from. For the standard configuration this is the holodeck/ folder, not the holodeck/bin folder. After you finish editing this file create a backup, as Tomcat will delete this file whenever you update the GW.

9.2. hibernate.properties

The hibernate.properties file is located in the conf/holodeck folder and defines additional database properties. The only property that has to be changed is "hibernate.dialect". Valid configuration properties are listed at http://docs.jboss.org/hibernate/orm/3.5/javadocs/org/hibernate/dialect/package-summary.html. If you are using an Oracle 11g database please use the org.hibernate.dialect.Oracle10gDialect.

```
hibernate.dialect=org.hibernate.dialect.DerbyDialect
hibernate.show_sql=false
hibernate.format_sql=true
hibernate.hbm2ddl.auto=none
```

Please contact your local database administrator for the configuration of the other properties. The default values should be fine, but your local policy may differ (i.e. on the number of allowed pooled connections).

9.3. PModes and public certificates

Copy the PMode file for your country from the configuration package downloaded in 5.3 to the folder specified in holodeck.xml (default: conf/holodeck/pmodes), and the security-config.xml file to the specified location (default: conf/holodeck/security-config.xml). The truststore.jks file sould be copied to conf/holodeck/keys, although its location can be configured in the security-config.xml.

9.4. Private key

Create a keystore containing only your private key for signing messages. This key should differ from the key you are using for SSL encryption. The keystore can be created using the standard Java "keytool" command. Documentation for this tool is available at http://docs.oracle.com/javase/1.4.2/docs/tooldocs/windows/keytool.html. Copy this keystore to the location configured in the security-config.xml file (default: conf/holodeck/keys/keystore.jks).

9.5. security-config.xml

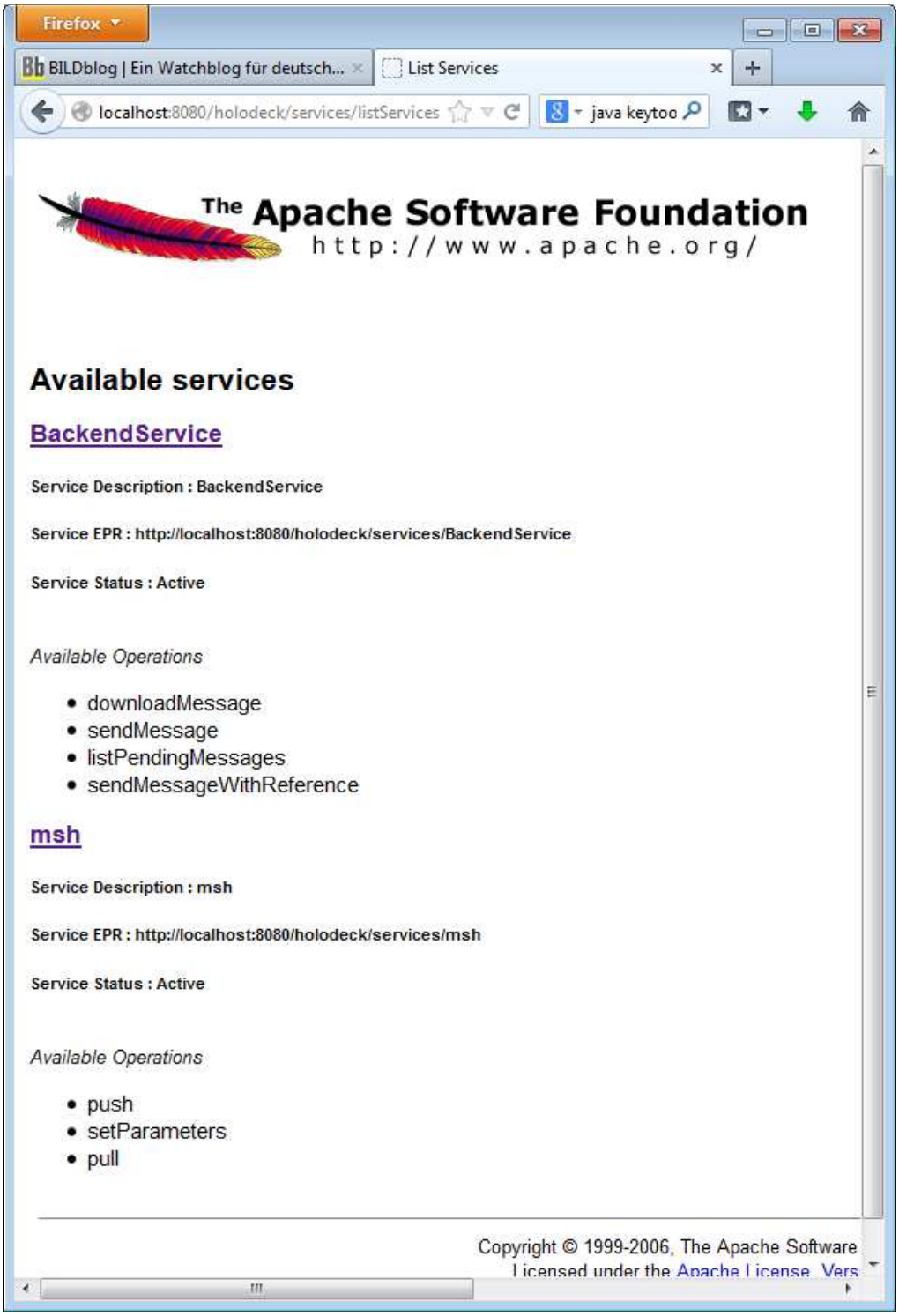
This file contains the keystore configuration and security agreements between participating countries. Please do edit contents inside the <tns:keystores> element only.

```
<tns:keystores>
	<tns:privateKeystore>
		<tns:localAlias>client</tns:localAlias>
		<tns:storepwd>apache</tns:storepwd>
		<tns:keypwd>apache</tns:keypwd>
		<tns:file>conf/holodeck/keys/keystore.jks</tns:file>
		<tns:storeType>jks</tns:storeType>
	</tns:privateKeystore>
	<tns:publicKeystore>
		<tns:storepwd>apache</tns:storepwd>
		<tns:file>conf/holodeck/keys/truststore.jks</tns:file>
		<tns:storeType>jks</tns:storeType>
	</tns:publicKeystore>
</tns:keystores>
```

key	description
tns:localAlias	The alias of your private key
tns:storepwd	The password of your keystore file
tns:keypwd	The password of your private key
tns:file	The location and name of your keystore file
tns:storeType	The type of the keystore file. Use jks keystores only

10. Start the server

To start the server execute .\bin\startup.bat or ./bin/startup.sh (depending on your operation system) from the servlet containers home folder ("holodeck", if you are using the full distribution package). STARTING DIRECTLY FROM THE bin/ FOLDER WILL RESULT IN FILENOTFOUNDEXCEPTIONS. To check if the GW is deployed correctly you can visit http://localhost:8080/holodeck/services/listServices (or use wget, if you are on a console-only server). The result should look like this:



This does not test your private/public keys, PModes and security configurations. Testing those requires the exchange of test messages with other GWs.

11. Further resources

11.1. Level of support

The level of support document contains agreements on the support lifecycle of the GW. It can be downloaded at BSCW.

11.2. e-CODEX JIRA Server

The e-CODEX JIRA server located at https://secure.e-codex.eu/jira contains the project's issue tracking system. If you have any problems regarding the installation please create a report using the "discussion" project.