imc
part of Scheer

Installation Instructions ILS

Windows / Tomcat 9 / IIS

imc's system solution historically known as CLIX *as of version 2013 is officially termed* ILS (imc Learning Suite*) instead. Some references to the "old" identifier may remain within this document, especially in the context of reproducing code snippets.*

im-c.com

Installation Instructions ILS

Windows / Tomcat 9 / IIS

Author(s): Dietmar Weinmann, Andreas Pohl
Date: 2020-11-23

Document	Description
Version	ILS 14.8
Status (Draft / Review / Finalisation)	Finalisation
Contact Person(s)	Andreas Pohl

History	Status	Who
2019-12-17	Review	Dietmar Weinmann
2020-10-02	Review	Andreas Pohl
2020-11-23	Finalisation	Dr. Peter Zönnchen

imc
information multimedia communication AG
Headquarter Saarbrücken
Scheer Tower, Uni-Campus Nord
D-66123 Saarbrücken
T. +49 681 9476-0 | Fax -530
info@im-c.com
im-c.com

Content

1 Important information about the installation

This document describes the installation of programs and required external components. The description is aimed at IT specialists. Specialist knowledge in the fields of databases, Java, web servers, servlet engines and server operation systems is required.

!	Description
!	**Replace example paths with actual paths!** Example drive and path information is used in this document for demonstration purposes. These details may be different in your own installation. For example, this document uses "`C:/appserver/instance/ils`" as the path to the ILS configuration directory. In your installation, the path could be "`H:/appserver/instance/ils`". Always replace these example paths with the actual paths in your installation!
!	**Replace placeholders and example parameter values with expedient values!** In this document, placeholders are often used for parameters or paths. The parameter values used are also merely examples. As every system environment is different, please replace placeholders and example parameter values with values which are adjusted to your system environment.

2 Installation steps at a glance

The following list will provide you with an overview of the required installation steps.

Action	Reference
Create application folder **Create a suitable folder structure.**	Section 3.1
Download and install external components **Download the JDK components from the Internet and install them in your folder structure.**	Section 3.2
Unpack ILS, ILP, Microservices, SOLR, IGS, LRS and PRS delivery packages **Extract the program, data and configuration files to the created folder structure.**	Section 3.3
Configure database **Install DBMS and carry out the connection test.**	Section 4.1
Path to the Java Development Kit (JDK) **Create the environment variable JAVA_HOME.**	Section 4.3
Configure ILS, ILP, Microservices, SOLR, IGS, LRS, PRS instance **This includes the adjustment of the configuration files `systemintegration.xml`, `web.xml` and `log4j.properties`.**	Sections 4.4, 0, 4.7, 4.8, 4.9, 4.10
Configure Servlet Engine Tomcat 9 **Integrate ILS, ILP, Microservices, SOLR and Gamification instance in Tomcat; Adjust ILS application contexts in the file `ils.xml` and `ils#data.xml`; adjust `wrapper.properties`, create connection to the database via JDBC driver; register Tomcat as a service. Furthermore, adjust the ILP application contexts in the file `ilp.xml`.**	Section 4.11
Test ILS application for the first time. **Start the ILS application for the first time; check the log file `clix.log`.**	Section 4.12
Configure web server IIS. **Create the connection to the Tomcat using the ISAPI connector is-api_redirect.dll; adjust the files `isapi_redirect.properties`, `workers.properties`, `uriworkermap.properties`; add the ISAPI filter to the IIS manager.**	Section 4.13
Access and test ILS in the browser. **Test the ILS application with upstream web server locally and from a remote computer.**	Section 4.14
Final steps: **Turn off directory browsing, remove example applications, change passwords and create license settings.**	Section 5

3 Prepare installation

3.1 Create appserver folder

Please extract the archives from setup.zip file to `C:\appserver\`

Unpack the install package

Copy the archive files **`apps.zip`**, **`instance.zip`**, **`data.zip`** and **`environment.zip`** from the install.zip to the created root directory (`C:\appserver\`) and extract them there.
After extracting the zip files your directory structure should look like this:

Folder structure	Content of folder
appserver	Level 1: Root directory, contains all components.
apps	Level 2: Contains all ILS, ILP and Solr (SPS) folder structure.
ils	Level 3: empty folder for ILS program files.
ilp	Level 3: empty folder for ILP program files.
microservices	Level 3: empty folder
solr	Level 3: empty folder for Solr as search tool.
igs	Level 3: empty folder for IGS program files.
lrs	Level 3: empty folder for LRS program files.
prs	Level 3: empty folder for PRS program files..
data	Level 2: empty folder for ILS data directory (learning content). For existing installations copy your content directory /data to the new installation directory
environment	Level 2: Contains all external components.
IIS_TC_connector_ila2013	Level 3: ISAPI connector (Empty folder)
java	Level 3: Java JDK with the JVM. (Empty folder)
jk	Level 3: Tomcat service
tomcat	Level 3: Servlet Engine Tomcat.

Folder structure	Content of folder
instance	Level 2: Contains folder structure for configuration files of SPS
ilp	Level 3: empty folder for configuration files of ILP
ils	Level 3: empty folder for configuration files of ILS
logs	Level 3: empty folder for application log files
solr	Level 3: empty folder for all configuration files of Solr
igs	Level 3: empty folder for all configuration files of IGS
lrs	Level 3: empty folder for all configuration files of LRS
prs	Level 3: empty folder for all configuration files of PRS

Table 1: Folder structure on the application server

3.2 Download external components

The following external components are required to operate the platforms which can be downloaded from the respective producers. Install the external components in the folder provided for this purpose (see section 3.1).
The external component versions supported by the platforms are listed in the systems prerequisites. We recommend using the newest sub-version within the respective used main version.

Java JDK (includes JVM)
http://www.oracle.com/technetwork/java/javase/downloads/index.html
Download the correct version for your operating system (32 bit or 64 bit).
The web server Internet Information Services (IIS) is also required for the operation of ILS and ILP. The IIS is already installed on Windows® server products.

3.3 Unpack ILS, ILP, SOLR, IGS, LRS and PRS delivery packages

The ILA package will be delivered either on a data carrier or on an imc AG SFTP server as a typical update package **`LS_<CUSTOM><Patchversion>_<Revision>.zip`** including the following files:

```
APP-FILES:      clix.war
                learningportal-<version>.war
                microservices-<version>.war
                gamificationService-<version>.war
                lrStore-<version>.war
                profileRecordStore-<version>.war
                ils-solr-<version>.war
DATA-FILE:      data.zip
CONF-FILES:     ils_conf.zip (ils_conf_<system>.zip)
                ilp_conf.zip (ilp_conf_<system>.zip), solr_conf.zip
                igs_conf.zip, prs_conf.zip, lrs_conf.zip
```

Extract the delivered program and configuration files according to the provided update advice.

Description	Archive and files	Comment
ILS program file	`clix.war`	Extract content to `.../apps/ils`
ILS configuration files	`ils_conf.zip`	Extract content to `.../instance/ils`
ILS data directory	`data.zip`	Copy to root directory of your content server (e.g. `c:/appserver`) and extract it there (with the option overwrite existing files and folder in `/data`)
ILP program file	`learningportal-<version>.war`	Extract content to `.../apps/ilp`
ILP configuration files	`ilp_conf.zip`	Extract content to `.../instance/ilp`
Microservices program file	`microservices-<version>.war`	Extract content to `.../apps/microservices`

Description	Archive and files	Comment
SOLR program file	`ils-solr-<version>.war`	Extract content to `.../apps/solr`
SOLR configuration files	`solr_conf.zip`	Extract content to `.../instance/solr`
IGS program file	`gamificationService-<version>.war`	Extract content to `.../apps/igs`
IGS configuration files	`igs_conf.zip`	Extract content to `.../instance/igs`
LRS program file	`lrStore-<version>.war`	Extract content to `.../apps/lrs`
LRS configuration files	`lrs _conf.zip`	Extract content to `.../instance/lrs`
PRS program file	`profileRecordStore-<version>.war`	Extract content to `.../apps/prs`
PRS configuration files	**`prs _conf.zip`**	Extract content to `.../instance/ prs`

Table 2: Application and configuration install paths

4 Installation

4.1 Prepare Database

4.1.1 Oracle

Oracle configuration

1. If ILS contains languages which require Unicode support (e.g. Chinese), an Oracle instance with the character set AL32UTF8 and NCHAR character set AL16UTF16 will be required.
2. Parameter PROCESSES should be set to the sum of the maximum size of the application server's connection pool of all nodes plus 25. E.g. if there are four nodes with a maximum value 200 each then PROCESSES should be set to 825.
3. Parameter MEMORY_TARGET should be set high enough to achieve a database cache hit ratio greater than 90%. It depends on the size of the database, usage pattern and number of concurrent users. It is recommended to use at least 2 GB and increase the value depending on the load and cache hit ratio.
4. In order to avoid the 4000-character limit for some ILS description fields the Oracle parameter MAX_STRING_SIZE=EXTENDED needs to be set. Follow the instructions in the Oracle reference manual for this parameter. Here is a summary how to set this parameter on a non-CDB:

   Go to directory `rdbms\admin` of the Oracle installation. The Oracle service needs to run and ORACLE_SID needs to point to the relevant Oracle instance. On Windows systems ORACLE_SID is set in the registry.

```
sqlplus / as sysdba;
SQL> shutdown;
SQL> startup upgrade;
SQL> alter system set max_string_size=extended;
SQL> @utl32k.sql
SQL> @utlrp.sql
SQL> shutdown;
SQL> startup;
SQL> quit
```

   If the execution of `@utl32k.sql` aborts with a message regarding elements in recycle bin you need to execute a "purge recyclebin" first and then continue with @utl32k.sql.

5. You may choose an existing tablespace or create a new own with any name for the ILS database. You may create a separate tablespace for indexes, but this is not required. To redirect indexes to a separate tablespace you need to set attribute indexTablespace of tag databaseServer in ILS configuration file systemintegration.xml to the name of the separate tablespace. It is recommended to use tablespaces in autoextend mode.

6. Create the required sorting order. Use the Oracle configuration parameter NLS_SORT for this purpose.
   The value GERMAN which is typically pre-installed on German Windows systems sorts according to the formula a-A-ä-b-B. The setting BINARY sorts according to the formula A-B-a-b-ä, i.e. it put upper case letters in front of lower case letters and places diacritical characters at the end of the list.
   The addition of _CI to a setting value of NLS_SORT ignores the upper case and lower case characters (case-insensitive).For example, BINARY_CI sorts according to the pattern a-A-b-B-ä.
   The addition of _AI to an NLS_SORT value ignores both upper case and lower case characters and the accents (case-insensitive and accent-insensitive).For example, BINARY_AI sorts according to the pattern ä-a-A-b-B. More details are available in the Oracle documentation.

Create the Oracle user for the ILS database

Create an Oracle user with the default table space of your choice (see following command which uses CLIX_DATA as an example) and assign a quota to the tablespace (we recommend unlimited to avoid unnecessary space problems). Allocate the roles connect, resource and create view to the user. The initial creation or import of the database can be done later.

```
create user ils_db identified by ils_pwd default tablespace CLIX_DATA
quota unlimited on clix_data;
grant connect, resource, create view to ils_db;
```

In case you use a second tablespace for indexes e.g. with name CLIX_IDX the command would look as.

```
create user ils_db identified by ils_pwd default tablespace CLIX_DATA
quota unlimited on clix_data quota unlimited on clix_idx;
grant connect, resource, create view to ils_db;
```

4.1.2 Microsoft SQL Server

SQL Server configuration

Ensure that the database server permits the SQL Server Authentication. To achieve this, select “SQL Server and Windows Authentication mode” in SQL Server the Security settings.

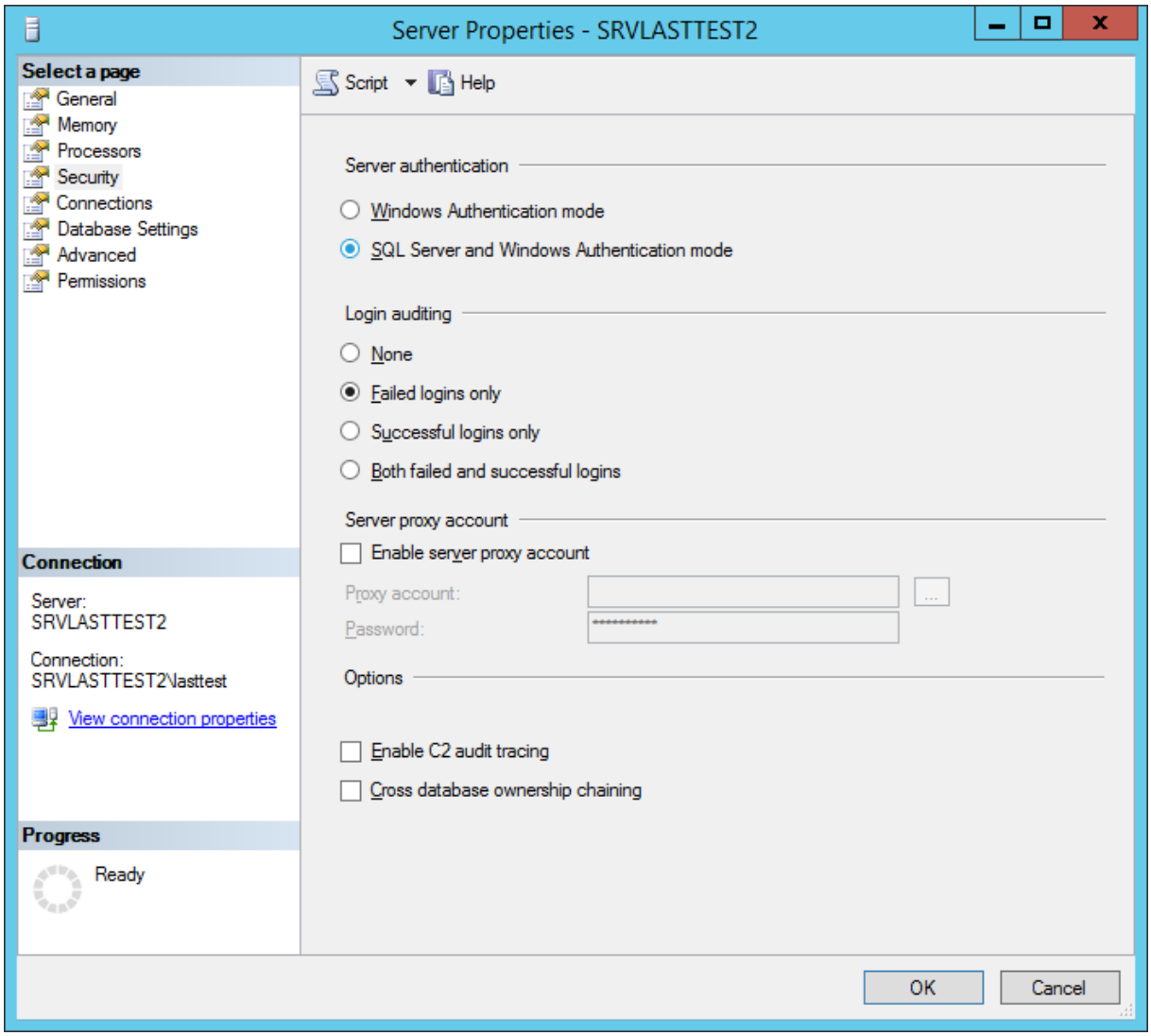

Fig. 4.1: MS SQL Server: Security settings in the “Server Properties”

Create a database for ILS

1. Create a database. Give the database a name, e.g. `ils_db`.
2. Add a login. Give the login a name, e.g. `ils_db` and select database `ils_db` as default database for this login.

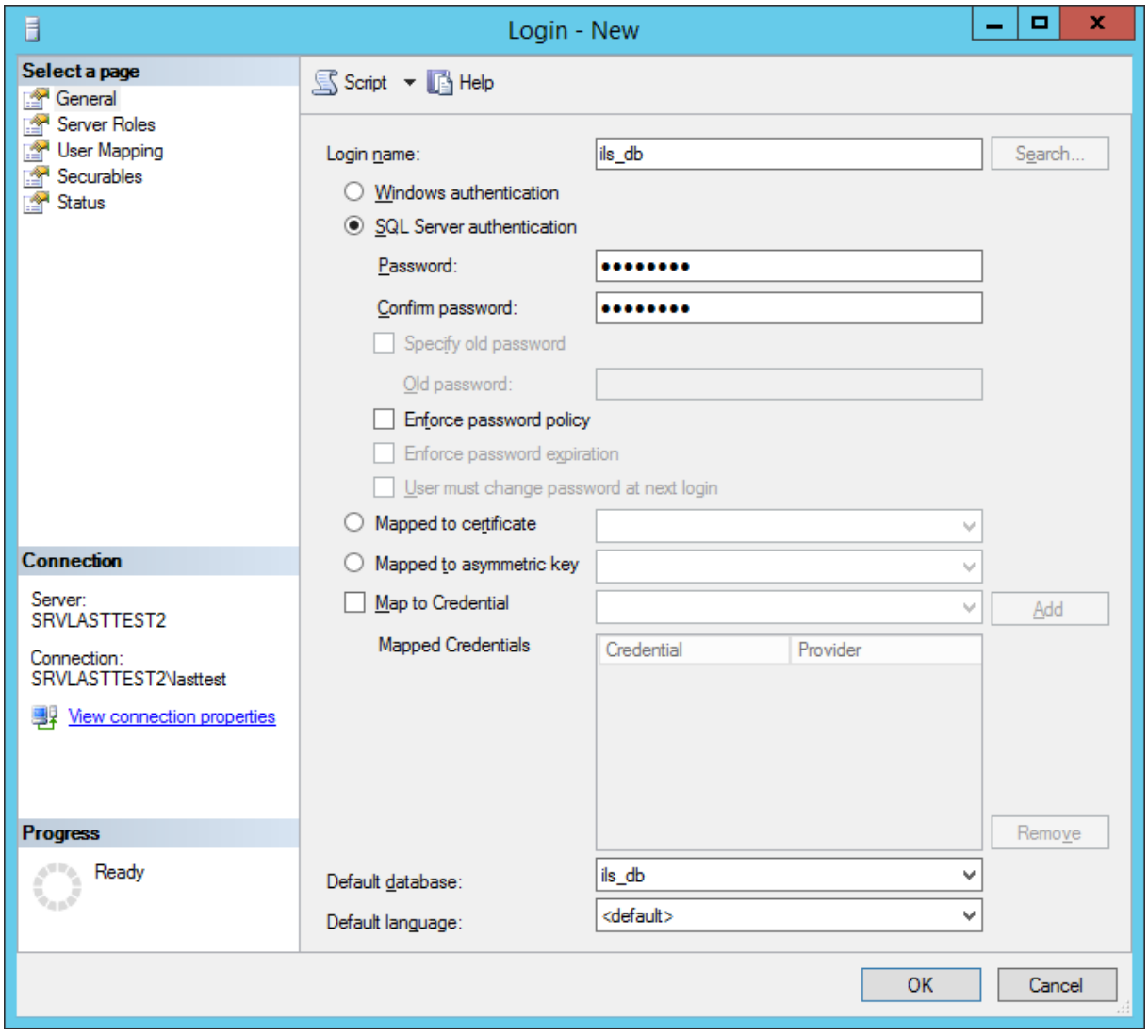

Fig. 4.2: MS SQL Server: Add a Login

3. Select page User Mapping and select database `ils_db` and role `db_owner` as shown in the following screen shot.

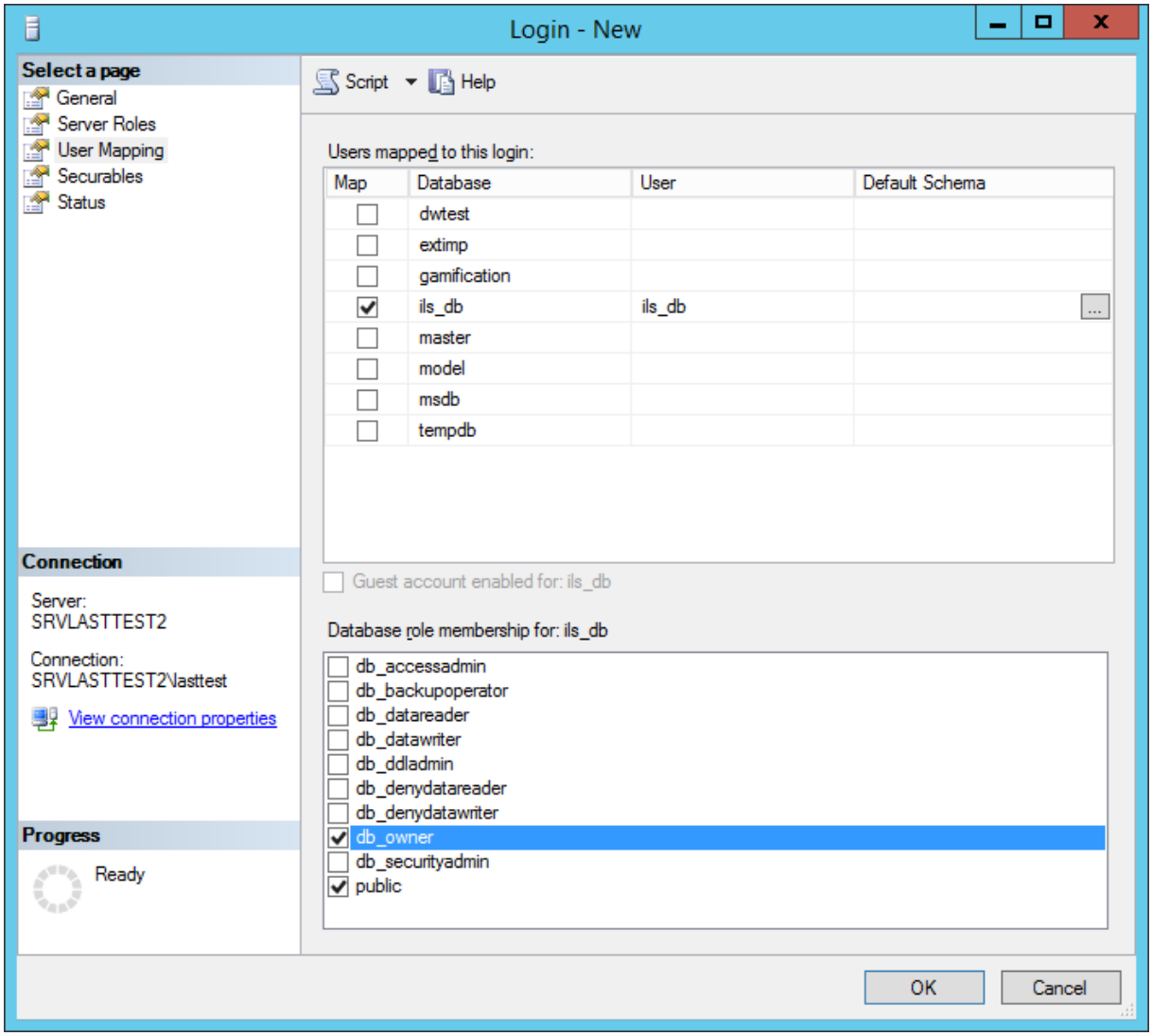

Fig. 4.3: MS SQL Server: User Mapping

4. Activate the SNAPSHOT-ISOLATION feature for the ILS database by executing the following two commands for the ILS database. You need to execute these statements on the master database. Note: If you have chosen the same name for database name and login you may skip this step since the system can do it automatically then.
This avoids database deadlocks, which can occur if there are several parallel ILS users and, in the worst cases, necessitate a restart of the application.

```
ALTER DATABASE clix SET ALLOW_SNAPSHOT_ISOLATION ON;
ALTER DATABASE clix SET READ_COMMITTED_SNAPSHOT ON;
```

4.1.3 PostgreSQL

PostgreSQL configuration

Make sure in configuration file pg_hba.conf there is an appropriate entry to allow connections from the application server. The following example will work for sure.

```
host  all  all 0.0.0.0/0 md5
```

In configuration file postgresql.conf set the following parameters:

1. Parameter MAX_CONNECTIONS should be set to the sum of the maximum size of the application server's connection pool of all nodes plus 25. E.g. if there are four nodes with a maximum value 200 each then MAX_CONNECTIONS should be set to 825.
2. Parameter SHARED_BUFFERS should initially be set to about 25% of the physical memory assuming it is a dedicated server just for PostgreSQL and therefore enough free memory is available. Based on the way PostgreSQL uses the operating system cache higher values might not be helpful.
3. Parameter EFFECTIVE_CACHE_SIZE should be set to half of the available physical memory. This parameter does not cause memory allocation of the specified size. It should rather be viewed as a hint to the query optimizer about available memory for caching (including operating system caching).
4. Parameter MAX_LOCKS_PER_TRANSACTION should be set to 100.
5. Parameter SHARED_PRELOAD_LIBRARIES should be set to 'pg_stat_statements' to be able to get SQL runtime statistics in case of performance problems. You also need to load this extension into the database as shown in the next paragraph.

Create an empty database for ILS

Create a database and give it a name, e.g. `ils_db`.

```
    create database ils_db with encoding UTF8;
```

Add a user e.g. `ils_db` and grant privileges to the user on the database.

```
    create user ils db with password 'imc clix';
    grant create, connect on database ils_db to ils_db;
```

Connect to the created database with system privileges and execute the following.

```
    create extension pg_stat_statements;
```

Restore the initial ILS database backup

With the initial delivery we also deliver a backup (SQL file) of the initial database. This backup needs to be restored into the empty database created in section 4.1.3 as shown here.

```
    psql -U ils_db -d ils_db -f ils_db.sql -q
```

4.2 Test Database Connection

An initial database test can be executed the most quickly with telnet on the database listener port. The default listener port is 1521 for Oracle, 1433 for SQL Server and 5432 for PostgreSQL. The database can alternatively be tested using database clients such as SQuirrel or DbVisualizer, where a JDBC driver can be integrated. Information about the correct JDBC driver for your database can be found in the document SystemRequirements. Information about the driver class and url settings can be found in section 4.11.3.

Carry out connection test with tool:

1. Install and launch the database client.
2. Configure the JDBC access to the database.
3. Check whether the connection can be made by creating a new table TESTINSTALL with an entry:

```
CREATE TABLE TESTINSTALL (TESTCOL VARCHAR(100));
INSERT INTO TESTINSTALL VALUES ('ils check 1-5');
```

4. Check whether a row can be read:

```
SELECT * FROM TESTINSTALL;
```

5. Delete the TESTINSTALL table:

```
DROP TABLE TESTINSTALL;
```

4.3 Java Development Kit

!	Description
!	**Replace example path information with actual paths!** The path information in this document is provided only as an example. Remember to always replace these example paths with the actual paths in your installation!

The environment variable `JAVA_HOME` needs to be set up for the operation of the Servlet Engine Apache Tomcat. `JAVA_HOME` provides the path to the Java Development Kit (JDK). Follow the following process:

Create the environment variable JAVA_HOME:

1. Select Start > System management > System > Advanced > Environment variables.
2. The **Environment variables** window will be displayed (see Fig. 4.7).
3. To create the new variable, click on **New** under **System variable**.
4. The **New system variable** window will be displayed.
5. Enter the text JAVA_HOME as the **Name of the variables**.
6. Enter the installation path to the JDK as the **Value of the variables**, e.g. C:/appserver/environment/jdk.
7. Click on **OK**.

The environment variable has been fully set up.

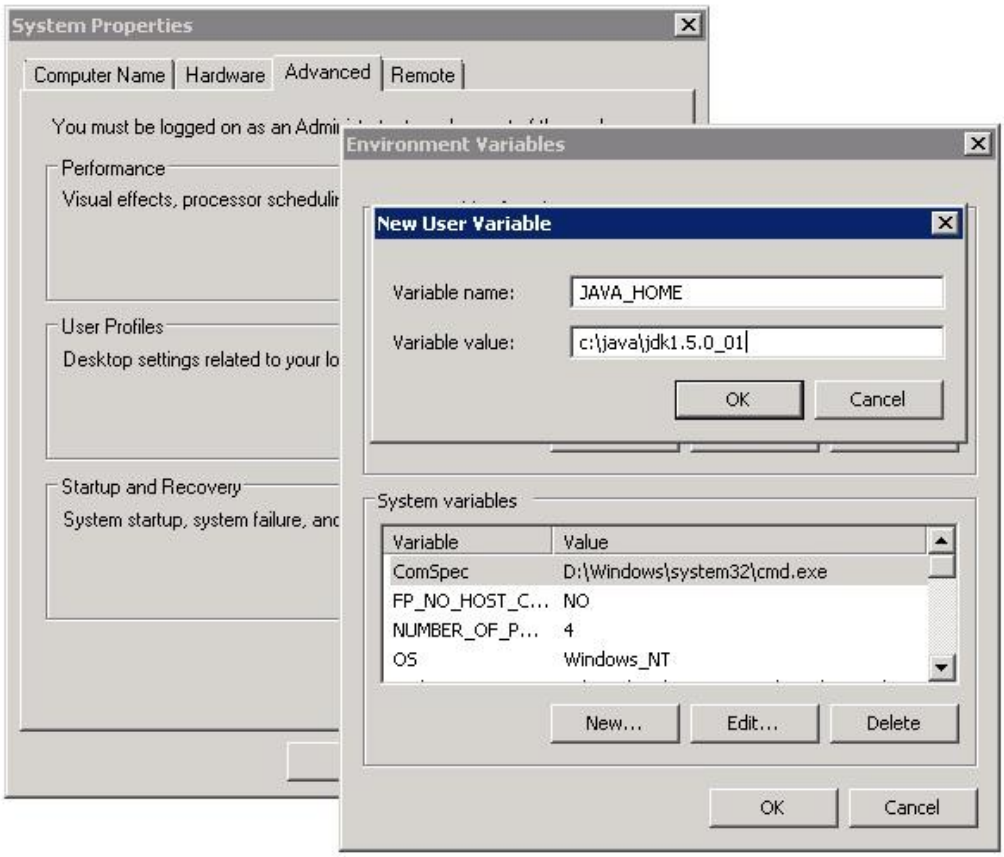

Fig. 4.4: Set up the environment variable

4.4 Prepare ILS instance

4.4.1 Adjust systemintegration.xml

The file **`systemintegration.xml`** specifies all configuration parameters which are relevant for the integration of ILS in your IT architecture. The parameters are described in the following table.

Adjust system integration.xml:

1. Open the file C:/appserver/instance/ils/systemintegration.xml with an editor.
2. Adjust the file according to your system environment. The parameters which need to be adjusted are identified in the file as Change_ME_ and explained in the following table.
3. Save the file.

!	Description
!	**Replace placeholders and example parameter values with expedient values!** In this document, placeholders are often used for parameters or paths. All parameters and URLs are examples. As every system environment is different, please replace placeholders and example parameter values with values which are adjusted to your system environment.

Parameter	Explanation	Example value
`<systemIntegrationsystemIdentifier="CHANGE_ME_SYSTEM_IDENTIFIER" … >`		
`systemIdentifier`	Provides the license key (identifier) of the installation and is used to identify the current installation and release the license settings confirmed by imc AG (e.g. “imc”).	
`CHANGE_ME_PORT_CLIX`	port of ILS (e.g. 80)	
`CHANGE_ME_PORT_TOMEE`	port of application server (e.g. 8080)	
`CHANGE_ME_DNS_CLIX`	DNS of `ils` (e.g. ils.yourcompany.com)	
`CHANGE_ME_ROOT_DIR`	CLIX root directory (e.g. `c:/appserver/`)	
`CHANGE_ME_DB_SCHEMA`	Database scheme (in mssql e.g.: `dbo`)	
`CHANGE_ME_JND_LOOKUP`	Tomcat: `java:comp/env/jdbc/clix` WebSphere: `jdbc/clix`	

Parameter	Explanation	Example value
CHANGE_ME_EMAIL_RECEIVER	Email address that receives all mails when email is in test mode (e.g. henry.meyer@yourcompany.com)	
CHANGE_ME_EMAIL_SENDER	Email address that sends mails from support backend (e.g. henry.meyer@yourcompany.com)	
CHANGE_ME_EMAIL_SERVERHOST	Email server (e.g. emailserver.yourcompany.com)	

Table 3: Adjust the systemintegration.xml file

4.4.2 Adjust path to log file (log4j.properties)

The file **log4j.properties** establishes the path to the log file.

Adjust file log4j.properties:

1. Open the file C:/appserver/instance/ils/log4j.properties with an editor.
2. Change CHANGE_ME_ROOT_DIR to C:/appserver/instance/logs/clix.log.
3. Save the file.

clix.log file will be created automatically after starting ILS.

4.5 Prepare ILP instance

This section describes the installation steps of imc learning portal (ILP).
Your ILP delivery will include the following parts:

Description	Archive and files	Comments
ILP	`ilp`	ILP Web application. which exits in the path `C:/appserver/apps/ilp`
ILP config	`Ilp` (configuration)	Configuration files for your ILP web application. Holds all relevant configuration files like: • **systemconfig.properties** • **log4j.properties** • **clientconfig.properties** • **navigation.xml** The path of ilp configuration file is `C:/appserver/instance/ilp`

4.5.1 Install ILP as web application

systemconfig.properties

The file `systemconfig.properties` specifies all configuration parameters which are relevant for the service communication between ILP other services

Adjust file system:

1. Open the file C:/appserver/instance/ilp/systemconfig.properties with an editor.
2. Adjust the file according to your system environment. The parameters which need to be adjusted are explained below,
3. Change CHANGE_ME_PORT_TOMEE to 8080,
4. Change CHANGE_ME_DNS_CLIX to your DNS name (e.g. learning.yourdomain.com),
5. Change CHANGE_ME_PORT_CLIX to 80 ,
6. Save the file.

log4j.properties

The file `log4j.properties` specifies all configuration parameters which are relevant for ILP.

Adjust file system:

1. Open the file C:/appserver/instance/ilp/log4j.properties with an editor.
2. Change `CHANGE_ME_ROOT_DIR` according to your system environment.
3. Save the file.

4.6 Prepare Microservices instance

This section describes the installation steps of the Microservices application. It is a bundle of smaller micro services taking tasks like Navigation or Course Desciption, etc.
Your delivery will include the following parts:

Description	Archive and files	Comments
Microservices	**`microservices`**	ILP Web application. which exits in the path **`C:/appserver/apps/microservices`**

4.6.1 Install Microservices as web application

systemconfig.properties

The file `systemconfig.properties` specifies all configuration parameters which are relevant for the service communication between ILP other services.

Adjust file:

1. Open the file C:/appserver/instance/ilp/systemconfig.properties with an editor.
2. Adjust the file according to your system environment. The parameters which need to be adjusted are explained below,
3. Check that `de.imc.services.url=/microservices/` already points to the relative URL of your microservices application or adapt to the patch / context that you would like to use for the microservices application
4. Save the file if you did any changes

systemintegration.xml

The file **`systemintegration.xml`** specifies all configuration parameters which are relevant for the integration of ILS in your IT architecture and the communication with other services.

Adjust file:

1. Open the file C:/appserver/instance/ils/systemintegration.xml with an editor.
2. Check that the URL in the following section at the end of the file holds the relative URL to your microservice application or adapt to the path / context that you would like to use for the microservices application.

```
<webServices>
          <webService name="microservices">
               <webServiceClient URL="/microservices/"/>
          </webService>
     </webServices>
```

3. Save the file.

4.7 Prepare Solr instance

This section describes the configuration steps of Solr.
Your ILP delivery will comprise the following parts:

Description	Archive and files	Comments
solr	`Solr`	Solr Platform, the Path of solar is `C:/appserver/apps/solr`
Solr config	`Solr` (Configuration)	Configuration files for your Solr platform. The path of the configuration files is `C:/appserver/instance/solr`

There is no further need for preparation at this point.

4.8 Prepare IGS instance

This section describes the installation steps of imc Gamification Service (IGS).
Your ILP delivery will include the following parts:

Description	Archive and files	Comments
IGS	`igs`	IGS Web application. which exits in the path `C:/appserver/apps/igs`
IGS config	`igs` (configuration)	Configuration files for your IGS web application. Holds all relevant configuration files like: • **systemconfig.properties** • **log4j.properties** The path of IGS configuration file is `C:/appserver/instance/igs`

4.8.1 Install IGS as web application

systemconfig.properties
The file `systemconfig.properties` specifies all configuration parameters which are relevant for IGS.

Adjust file system:

1. Open the file C:/appserver/instance/igs/systemconfig.properties with an editor.
2. Adjust the file according to your system environment. The parameters which need to be adjusted are explained below.
3. Change CHANGE_ME_PORT_TOMEE to 8080
4. Set `de.imc.igs.core.igs.service.endpoint.publicUrl` to the public URL of your IGS installation (e.g. http://learning.yourdomain.com/igs). This is required to see images of gamification objects (badges) in your ILS installation.
5. Save the file.

log4j.properties
The file **`log4j.properties`** specifies all configuration parameters which are relevant for IGS.

Adjust file system:

1. Open the file C:/appserver/instance/igs/log4j.properties with an editor.

2. Change CHANGE_ME_ROOT_DIR according to your system environment.
3. Save the file.

4.9 Prepare LRS instance

This section describes the installation steps of imc Learning Recordstore (LRS).
Your ILP delivery will include the following parts:

Description	Archive and files	Comments
LRS	`lrs`	LRS Web application. which exits in the path `C:/appserver/apps/lrs`
LRS config	`lrs` (configuration)	Configuration files for your LRS web application. Holds all relevant configuration files like: • **systemconfig.properties** • **log4j.properties** The path of LRS configuration file is `C:/appserver/instance/lrs`

4.9.1 Install LRS as web application

systemconfig.properties

The file `systemconfig.properties` specifies all configuration parameters which are relevant for IGS.

Adjust file system:

1. Open the file C:/appserver/instance/lrs/systemconfig.properties with an editor.
2. Adjust the file according to your system environment. The parameters which need to be adjusted are explained below
3. Change CHANGE_ME_PORT_TOMEE to 8080.
4. Save the file.

log4j.properties

The file `log4j.properties` specifies all configuration parameters which are relevant for LRS.

Adjust file system:

1. Open the file C:/appserver/instance/lrs/log4j.properties with an editor.
2. Change `CHANGE_ME_ROOT_DIR` according to your system environment.
3. Save the file.

4.10 Prepare PRS instance

This section describes the installation steps of imc Profile Recordstore (PRS).
Your ILP delivery will include the following parts:

Description	Archive and files	Comments
PRS	`prs`	PRS Web application. which exits in the path `C:/appserver/apps/prs`
PRS config	`prs` (configuration)	Configuration files for your PRS web application. Holds all relevant configuration files like: • **systemconfig.properties** • **log4j.properties** The path of PRS configuration file is `C:/appserver/instance/prs`

4.10.1 Install PRS as web application

systemconfig.properties

The file `systemconfig.properties` specifies all configuration parameters which are relevant for IGS.

Adjust file system:

1. Open the file C:/appserver/instance/prs/systemconfig.properties with an editor.
2. Adjust the file according to your system environment. The parameters which need to be adjusted are explained below.
3. Change CHANGE_ME_PORT_TOMEE to 8080.
4. Save the file.

log4j.properties

The file `log4j.properties` specifies all configuration parameters which are relevant for PRS.

Adjust file system:

1. Open the file C:/appserver/instance/prs/log4j.properties with an editor.
2. Change `CHANGE_ME_ROOT_DIR` according to your system environment.
3. Save the file.

4.11 Configure Tomcat 9

!	Description
!	Replace example path information with actual paths! The path information in this document is provided only as examples. Remember to always replace these example paths with the actual paths in your installation!

4.11.1 Adjust server.xml

To allow the servlet engine to communicate correctly with the ILS web application, free ports are needed (i.e. as shutdown or connector port)
You can adjust the example file and then add it to the configuration directory for the ILS application in the servlet engine:
`C:/appserver/environment/tomcat/conf/server.xml`.

Adjust `server.xml` file

1. Open the `server.xml` file in an editor. Adjust the file as described below. Check the values printed in bold and adjust the values only where required if the ports are already in use. The values are explained in the following Table below.
2. Replace placeholders and example paths with the actual paths.
3. Save the `server.xml` in the ILS instance configuration directory, e.g. `C:/appserver/environment/tomcat/conf/server.xml`

Extract from the file `server.xml`:

```
<Server port="8005" shutdown="SHUTDOWN">
   ...
<Service name="Catalina">
   ...
<Connector port="8080" protocol="HTTP/1.1" connectionTimeout="20000"
redirectPort="8443" />
   ...
<Connector port="8009" maxThreads="1200" enableLookups="false" redi-
rectPort="8443"
 protocol="AJP/1.3"/>

<Engine name="Catalina" defaultHost="localhost">
   ...
<Host name="localhost" appBase="webapps" unpackWARs="false"
 autoDeploy="false">
   ...
   ...
</Host>
</Engine>
</Service>
</Server>
```

The parameters are explained in detail in the following table:

Parameter	Explanation	Example value
<Server port="8005" shutdown="SHUTDOWN">		
Port	The shutdown port must match the shutdown port in the file `wrapper.properties` (see section *0*)! The file can be found at `../appserver/environment/jk/`	8005
<Connector port="8009" maxThreads="1200" enableLookups="false" redirectPort="8443" protocol="AJP/1.3"/>		
Port	TCP port of the connector used by the servlet engine and web server to report with each other. Must match the port information in the file `workers.properties` (see section *4.13.1*)! The file can be found at `../appserver/environment/IIS_TC_connector_ila2013/conf/`	8009
<Connector port="8080" protocol="HTTP/1.1" connectionTimeout="20000" redirectPort="8443" />		
Port	Internal HTTP port of the Tomcat. Must match the port information (`CHANGE_ME_PORT_TOMEE`) in the file `systemintegration.xml` (see section *4.4.1*)! And it must match the port information (`CHANGE_ME_PORT_TOMEE`) in the file `systemconfig.properties` (see section *0*)!	8080
Optional: Use the following setting only in case you are using only HTTPS as public URL of your server. To enhance security (see https://www.owasp.org/index.php/Session_Management_Cheat_Sheet#Secure_Attribute) add secure="true" to the connector setting: `<Connector port="8080" protocol="HTTP/1.1" connectionTimeout="20000" redirectPort="8443" secure="true"/>`		

4.11.2 Adjust context.xml

To enhance security (see https://www.owasp.org/index.php/HttpOnly) open the **context.xml** file in an editor and replace **<Context>** with **<Context useHttpOnly="true">**.

4.11.3 Adjust ils.xml and ils#data.xml

The application contexts "**ils**" and "**data**" are needed for ILS.
"data" refers to the data directory used by ILS.
The definition of the contexts is controlled in the Tomcat configuration file **ils.xml**. The file **ils.xml** is included in the pre-configured example files of the ILS delivery package (see 3.3).
You can adjust the example file and then add it to the configuration directory for the ILS application in the servlet engine:
C:/appserver/environment/tomcat/conf/Catalina/localhost/ils.xml.

Adjust ils.xml file

1. Open the ils.xml file in an editor. The file will be located under the pre-configured ILS example files.
2. Adjust the file as described below. Check the values starting with CHANGE_ and adjust the values where required. The values are explained in the following Table below.
3. Replace placeholders and example paths with the actual paths.
4. Save the ils.xml in the Tomcat instance configuration directory, e.g. C:/appserver/environment/tomcat/conf/Catalina/localhost/ils.xml.

Extract from the file ils.xml (SQL Server example):

```
<?xml version="1.0" encoding="UTF-8"?>
<Context path="ils" docBase="C:/appserver/apps/ils " re-
loadable="false" unpackWAR="false" sessionCookieName="ils_JSESSIONID"
>

<Resource
    auth="Container"
    name="jdbc/clix"
    type="javax.sql.DataSource"
    factory="org.apache.tomcat.jdbc.pool.DataSourceFactory"
    driverClassName="com.microsoft.sqlserver.jdbc.SQLServerDriver"
    url="jdbc:sqlserver://CHANGE_ME_DB_SERVER:1433;Database-
Name=CHANGE_ME_DBNAME"
    username="CHANGE_ME_DBUSER"
    password="CHANGE_ME_DBPASSWD"
    maxWait="-1"
    maxActive="600”
    validationQuery="select 1"
    testWhileIdle="true"
    timeBetweenEvictionRunsMillis="30000"
/>

<Parameter name="clix.configpath" value="C:/appserver/instance/ils/"
override="false"/>
</Context>
```

Note: For **ORACLE** only also add the following line in the Resource tag:

```
connectionProperties="defaultRowPrefetch=100"
```

The parameters are explained in detail in the following table:

Parameter	Explanation	Example value
Context.path	Enter the context path entered into the configuration file `systemintegration.xml` (parameter `applicationCluster.contextPath`), see section 4.4.1. If the context path is empty or equals "/", leave the attribute empty: `<Context path="" ...>`	`/ils`
Context.docBase	Provide the physical path to the ILS program directory entered into the configuration file `systemintegration.xml` (parameter `applicationCluster.applicationServer.physicalPath`), see section 4.4.1, e.g. `C:/appserver/apps/ils`	`C:/appserver/apps/ils`
Resource .name	JNDI descriptor for the data source used. Must match the last two parts of the parameter `JNDILookupString` from the `systemintegration.xml` (see section 4.4.1).	`jdbc/ils`
Resource.factory	With this entry the new multi-threading Tomcat Connection Pool is activated.	
Resource .driverClassName	Driver class of the JDBC driver. Oracle: `oracle.jdbc.driver.OracleDriver` MSSQL: `com.microsoft.sqlserver.jdbc.SQLServerDriver` PostgreSQL: `org.postgresql.Driver`	

Resource .url	URL of the database server. The IP address `192.168.0.7` stands for the IP of the database server; port 1521, 1433 or 5432 stands for the listener port. imc stands for the Oracle SID.
	Oracle:
	`jdbc:oracle:thin:@192.168.0.7:1521:IMC`
	MSSQL:
	`jdbc:sqlserver://192.168.0.7:1433;DatabaseName=ils_db`
	PostgresSQL:
	`jdbc:postgresql://192.168.0.7:5432/ils_db?autosave=always`	

Resource .username	Name of the database user e.g. `ils_db`.	

Parameter	Explanation	Example value
Resource .password	Password for the database user.	
maxWait	The maximum number of milliseconds that the pool will wait (when there are no available connections) for a connection to be returned before throwing an exception, or -1 to wait indefinitely	
maxActive	The maximum number of database connections of the Tomcat connection pool. As a rule of thumb, this parameter should be set to half the value of `maxThreads`.	
validationQuery	The `validationQuery` configuration ensures that only valid database connections are returned from the Tomcat connection pool. Oracle: select 1 from dual MSSQL or PostgreSQL: select 1	
testWhileIdle timeBetweenEvictionRunsMillis	These are additional parameters for the `validationQuery` with fixed values as shown above.	
connectionProperties	Allows to set additional connection properties. Currently it is only used for Oracle. Oracle: "`defaultRowPrefetch=100`" For Oracle the default fetch size is just 10. Performance tests showed about 10% improvement by setting it to 100.	
Parameter.value	The path of configuration files of ILS	`C:/appserver/instance/ils`

Adjust `ils#data.xml` file

1. Open the `ils#data.xml` file in an editor. The file will be located under the pre-configured ILS example files.
2. Adjust the file as described below. Check the values printed in bold and adjust the values where required. The values are explained in the following Table 4.Replace placeholders and example paths with the actual paths.
3. Save the file in the ILS instance configuration directory, e.g. C:/appserver/environment/tomcat/conf/Catalina/localhost/ils#data.xml

Extract from the file `ils#data.xml`:

```
<?xml version="1.0" encoding="UTF-8"?>
<Context path="/ils/data" docBase="C:/appserver/data" re-
loadable="false" >
</Context>
```

Parameters are explained in detail in the following table:

Parameter	Explanation	Example value
Context.path	Enter the context path as an alias of physical path	`/ils/data`
Context.docBase	Provide the physical path to the data folder	`C:/appserver/data`

4.11.4 Integrate database driver

ILS contacts the database via a JDBC driver. This driver must have access to the servlet engine. For this purpose, the JDBC driver is copied into the subdirectory provided in the Tomcat installation.

Integrate JDBC driver:

1. Depending on the database used get the appropriate JDBC driver as described in document System_Requirements.pdf.
2. Copy the driver to the subdirectory lib of the Tomcat installation, e.g. `C:/appserver/environment/tomcat/lib`
   → The database driver has been integrated.

4.11.5 Adjust wrapper.properties

The file wrapper.properties is provided in the ILS delivery package and provides the pre-setting and JVM start parameters for the Windows service (Tomcat), which is set up in the following segment. The file must be adjusted according to your installation environment.

Adjust `wrapper.properties` file

1. Add the new subdirectory jk to the environment installation directory, if it does not exist e.g. `C:/appserver/environment/jk`.
2. Copy the files `wrapper.properties` into the `jk` directory you have just created.
3. Open the file in an editor.
4. Adjust the file as described in the example below. Check the values printed in bold. Replace example paths with the actual paths.

Typical configuration of the `wrapper.properties` file

```
# Path to JDK and Tomcat
wrapper.tomcat_home= C:/appserver/environment/tomcat
wrapper.java_home= C:/appserver/environment/jdk

# Path to the Tomcat log files; in this example, the
# Tomcat log files are saved in the ILS log directory.
wrapper.stdout= C:/appserver/instance/logs/stdout.log
wrapper.stderr= C:/appserver/instance/logs/stderr.log

wrapper.ld_path=$(wrapper.java_home)/jre/bin
wrapper.ld_path=$(wrapper.tomcat_home)/bin/native
wrapper.class_path=$(wrapper.tomcat_home)/bin/bootstrap.jar
wrapper.class_path=$(wrapper.java_home)/lib/tools.jar

wrapper.javabin=$(wrapper.java_home)/bin/java.exe
wrapper.startup_class=org.apache.catalina.startup.Bootstrap

# Path to Tomcat policy, to file server.xml,
# Information about shutdown port from server.xml
wrapper.tomcat_policy= C:/appserver/environment/tomcat/
conf/catalina.policy
wrapper.server_xml= C:/appserver/environment/tomcat/conf/server.xml
wrapper.shutdown_port=8005

#Adjust heap memory, see below for explanation
wrapper.shutdown_protocol=ajp13
wrapper.cmd_line=$(wrapper.javabin) -Xrs
-Djava.security.policy="$(wrapper.tomcat_policy)"
-Dtomcat.home="$(wrapper.tomcat_home)"
-classpath $(wrapper.class_path)
-Xms2700m -Xmx2700m -server -verbose:gc -Duser.timezone=Europe/Berlin
-XX:+PrintGCDetails -Xloggc:C:/appserver/instance/logs/gc.log
$(wrapper.startup_class) -config $(wrapper.server_xml) start
```

Additional property value for IGS

MSSQL is considered default for IGS , to use Oracle or PostgreSQL along with IGS please add the following value to the wrapper.cmd_line

```
"-Dhibernate.dialect=de.imc.igs.core.db.OracleUnicodeDialect"
```

or

```
"-Dhibernate.dialect=de.imc.igs.core.db.PostgreSQLUnicodeDialect"
```

Adjust heap memory

See Systemrequirements.doc to change memory settings according to the number of concurrent users.

4.11.6 Adjust ilp.xml

The servlet engine needs to communicate the ILP application that is why the host need to be configured with help of the application contexts, this latter provides the paths to the relevant directories.

Adjust `ilp.xml` file

1. Open the `ilp.xml` file in an editor. The file will be located under the pre-configured ILS example files.
2. Adjust the file as described below. Check the values printed in bold and adjust the values where required. The values are explained in the following table.
3. Replace placeholders and example paths with the actual paths.
4. Save the ilp.xml in the Tomcat instance configuration directory, e.g. C:/appserver/environment/tomcat/conf/Catalina/localhost/ilp.xml

Extract from the file `ils.xml`:

```
<?xml version="1.0" encoding="UTF-8"?>
<Context path="/ilp" docBase="C:/appserver/apps/ilp" re-
loadable="false" sessionCookieName="ilp JSESSIONID">
<Parameter
     name="de.imc.ila.core.CONF_DIR"
     value="c:/appserver/instance/ilp"
     override="false"/>
</Context>
```

The parameters are explained in detail in the following table:

Parameter	Explanation	Example value
Context.path	Enter the context path of ILP.	`/ilp`
Context.docBase	Provide the physical path to the ILP program.	`C:/appserver/apps/ilp`
Parameter.value	This parameter provide the configuration files path.	`c:/appserver/instance/ilp`

Table 4: Parameters of the file `ilp.xml`

4.11.7 Adjust microservices.xml

The servlet engine needs to communicate the ILP application that is why the host need to be configured with help of the application contexts, this latter provides the paths to the relevant directories.

Adjust `microservices.xml` file

1. Open the `microservices.xml` file in an editor. The file will be located under the pre-configured ILS example files.
2. Adjust the file as described below. Check the values printed in bold and adjust the values where required. The values are explained in the following table.
3. Replace placeholders and example paths with the actual paths.
4. Save the microservices.xml in the Tomcat instance configuration directory, e.g. C:/appserver/environment/tomcat/conf/Catalina/localhost/microservices.xml

Extract from the file `microservices.xml`:

```
<?xml version='1.0' encoding='utf-8'?>
<Context reloadable="false"
 sessionCookieName="microsrv_JSESSIONID"
 docBase="c:/appserver/apps/microservices"/>
```

The parameters are explained in detail in the following table:

Parameter	Explanation	Example value
Context.docBase	Provide the physical path to the microservices program.	`C:/appserver/apps/microservices`

Table 4: Parameters of the file `microservices.xml`

4.11.8 Adjust solrsrv.xml

Solr is a search platform which is used for indexing and search the data of ILP and ILS trough Tomcat, for this reason a new context must be created in order to provide the directories of Solr platform and configuration files.

Adjust `solrsrv.xml` file

1. Open the `solrsrv.xml` file in an editor. The file will be located under the pre-configured Solr example files.
2. Adjust the file as described below. Check the values printed in bold and adjust the values where required. The values are explained in the following Table.
3. Replace placeholders and example paths with the actual paths.
4. Save the solrsrv.xml in Tomcat configuration directory, e.g. C:/appserver/environment/tomcat/conf/Catalina/localhost/solrsrv.xml

Extract from the file `solrsrv.xml`:

```
<?xml version="1.0" encoding="UTF-8"?>
<Context path="/solrsrv" docBase="C:/appserver/apps/solr" re-
loadable="false" sessionCookieName="solr_JSESSIONID" >
     <Environment override="false" value="C:/appserver/instance/solr"
type="java.lang.String" name="solr/home"/>
     <Parameter name="restApiUrl" value="http://localhost:8080/ils/re-
stapi" override="false"/>
</Context>
```

The parameters are explained in detail in the following table:

Parameter	Explanation	Example value
Context.path	Enter the context path into the configuration file `solrsrv.xml`	`/solrsrv`
Context.docBase	Provide the physical path to the Solr Application	`C:/appserver/apps/solr`
Parameter.value	This parameter provide the URL of ils rest api	http://localhost:8080/ils/restapi

4.11.9 Adjust igs.xml

The servlet engine needs to communicate the IGS application that is why the host need to be configured with help of the application contexts, this latter provides the paths to the relevant directories.

Adjust `igs.xml` file

1. Open the `igs.xml` file in an editor. The file will be located under the pre-configured ILS example files.
2. Adjust the file as described below. Check the values printed in bold and adjust the values where required. The values are explained in the following table.
3. Replace placeholders and example paths with the actual paths.
4. Save the `igs.xml` in the Tomcat instance configuration directory, e.g.

```
C:/appserver/environment/tomcat/conf/Catalina/localhost/igs.xml
```

Extract from the file `igs.xml` (SQL Server example):

```
<Context path="/igs" docBase=" C:/appserver/apps/igs/" re-
loadable="false" sessionCookieName="gams_JSESSIONID">
<Parameter   name="de.imc.igs.core.CONF_DIR"     value=" C:/apps-
erver/instance/igs/" override="false"/>
<Resource
    auth="Container"
    name="jdbc/experienceapi"
    type="javax.sql.DataSource"
    factory="org.apache.tomcat.jdbc.pool.DataSourceFactory"
    driverClassName="com.microsoft.sqlserver.jdbc.SQLServerDriver"
    url="jdbc:sqlserver://CHANGE_ME_DB_SERVER:1433;Database-
Name=CHANGE_ME_DBNAME"
    username="CHANGE_ME_DBUSER"
    password="CHANGE_ME_DBPASSWD"
    maxWait="-1"
    maxActive="600”
    validationQuery="select 1"
    testWhileIdle="true"
    timeBetweenEvictionRunsMillis="30000"
/>
</Context>
```

Note: For **ORACLE** only also add the following line in the Resource tag:

```
connectionProperties="defaultRowPrefetch=100"
```

The parameters are explained in detail in the following table:

Parameter	Explanation	Example value
Context.path	Enter the context path of IGS	`/igs`
Context.docBase	Provide the physical path to the IGS program.	`C:/appserver/apps/igs`
Parameter.value	This parameter provide the configuration files path.	`C:/appserver/instance/igs`

Table 5: Parameters of the file `igs.xml`

4.11.10 Adjust lrs.xml

The servlet engine needs to communicate the LRS application that is why the host need to be configured with help of the application contexts, this latter provides the paths to the relevant directories.

Adjust `lrs.xml` file

1. Open the `lrs.xml` file in an editor. The file will be located under the pre-configured ILS example files.
2. Adjust the file as described below. Check the values printed in bold and adjust the values where required. The values are explained in the following table.
3. Replace placeholders and example paths with the actual paths.
4. Save the `lrs.xml` in the Tomcat instance configuration directory, e.g.

```
C:/appserver/environment/tomcat/conf/Catalina/localhost/lrs.xml
```

Extract from the file `lrs.xml` (SQL Server example):

```
<Context path="/lrs" docBase=" C:/appserver/apps/lrs/" re-
loadable="false" sessionCookieName="gams_JSESSIONID">
<Parameter   name="de.imc.lrs.core.CONF_DIR"     value=" C:/apps-
erver/instance/lrs/" override="false"/>
<Resource
    auth="Container"
    name="jdbc/recordstoreapi"
    type="javax.sql.DataSource"
    factory="org.apache.tomcat.jdbc.pool.DataSourceFactory"
    driverClassName="com.microsoft.sqlserver.jdbc.SQLServerDriver"
url="jdbc:sqlserver://CHANGE_ME_DB_SERVER:1433;Database-
Name=CHANGE_ME_DBNAME"
    username="CHANGE_ME_DBUSER"
    password="CHANGE_ME_DBPASSWD"
    maxWait="-1"
    maxActive="600”
    validationQuery="select 1"
    testWhileIdle="true"
    timeBetweenEvictionRunsMillis="30000"
/>
</Context>
```

Note: For **ORACLE** only also add the following line in the Resource tag:
`connectionProperties="defaultRowPrefetch=100"`

The parameters are explained in detail in the following table:

Parameter	Explanation	Example value
Context.path	Enter the context path of LRS	`/lrs`
Context.docBase	Provide the physical path to the LRS program.	`C:/appserver/apps/lrs`
Parameter.value	This parameter provide the configuration files path.	`c:/appserver/instance/lrs`

Table 6: Parameters of the file `lrs.xml`

4.11.11 Adjust prs.xml

The servlet engine needs to communicate the PRS application that is why the host need to be configured with help of the application contexts, this latter provides the paths to the relevant directories.

Adjust `prs.xml` file

1. Open the `prs.xml` file in an editor. The file will be located under the pre-configured ILS example files.
2. Adjust the file as described below. Check the values printed in bold and adjust the values where required. The values are explained in the following table.
3. Replace placeholders and example paths with the actual paths.
4. Save the prs.xml in the Tomcat instance configuration directory, e.g. C:/appserver/environment/tomcat/conf/Catalina/localhost/prs.xml

Extract from the file `prs.xml` (SQL Server example):

```
<Context path="/prs" docBase=" C:/appserver/apps/prs/" re-
loadable="false" sessionCookieName="gams_JSESSIONID">
<Parameter   name="de.imc.prs.core.CONF_DIR"     value=" C:/apps-
erver/instance/prs/" override="false"/>
<Resource
    auth="Container"
    name="jdbc/profilestore"
    type="javax.sql.DataSource"
    factory="org.apache.tomcat.jdbc.pool.DataSourceFactory"
    driverClassName="com.microsoft.sqlserver.jdbc.SQLServerDriver"
url="jdbc:sqlserver://CHANGE_ME_DB_SERVER:1433;Database-
Name=CHANGE_ME_DBNAME"
    username="CHANGE_ME_DBUSER"
    password="CHANGE_ME_DBPASSWD"
    maxWait="-1"
    maxActive="600”
    validationQuery="select 1"
    testWhileIdle="true"
    timeBetweenEvictionRunsMillis="30000"
/>
</Context>
```

Note: For **ORACLE** only also add the following line in the Resource tag:

```
connectionProperties="defaultRowPrefetch=100"
```

The parameters are explained in detail in the following table:

Parameter	Explanation	Example value
Context.path	Enter the context path of PRS	`/prs`
Context.docBase	Provide the physical path to the PRS program.	`C:/appserver/apps/prs`
Parameter.value	This parameter provide the configuration files path.	`c:/appserver/instance/prs`

Table 7: Parameters of the file `prs.xml`

4.11.12 Validation Query

When ILS/IGS is operated with a database, this may result in problems with invalid database connections in the Tomcat database connection pool. To avoid such problems, we recommend supplementing the parameterization of the data source as described in the <Resource> element of the configuration with the following parameters. In case of Oracle the validation query needs to be changed to “select 1 from dual”.

It is suggested to enable validation query string after you have the ILS initialized after the first start.

```
...
validationQuery="select 1"
testWhileIdle="true"
timeBetweenEvictionRunsMillis="30000"
/>
```

!	Description
!	A detailed explanation of the Tomcat connection pool parameters is provided in the Tomcat documentation: https://tomcat.apache.org/tomcat-9.0-doc/jdbc-pool.html.

4.11.13 Register Tomcat as a service

!	Description
!	**Set the Unicode support BEFORE starting Tomcat for the first time!** If the database is empty, ILS will carry out an automatic initial upgrade before the Tomcat service is launched for the first time. If ILS needs to support languages which require Unicode support by the database (e.g. Greek or Arabic), set the attribute Unicode Database in the file systemintegration.xml to true before the initial upgrade (see section 4.4.1).

Once the launch parameters of the Tomcat service have been defined in the file `wrapper.properties` (see section 4.11.5), the servlet engine can register Tomcat as a Windows service. From Windows Server 2008, you will need to be logged in as an administrator to create a service.

Install and launch Windows service `tomcat_ils`:

1. If a Tomcat service has already been installed and launched, deactivate this server before the installation of Tomcat.
2. Copy the file **`jk_nt_service.exe`** from the ILS installation directory into the directory `C:/appserver/environment/jk`.
3. Change to the Windows entry request.
4. Change to the `jk` directory here (see following description).
5. Request the executable installation file **`jk_nt_service.exe`** with the option `-i`, providing the service name (e.g. `tomcat_ils`) and the configuration file `wrapper.properties`.

```
CD C:\appserver\environment\jk\
jk_nt_service

To install the service:
jk_nt_service.exe -i <service name> {optional params} <config proper-
ties file>
    Optional parameters
        -u <user name> - In the form DomainName\UserName (.\UserName
for local)
        -p <user password>
        -a - Set startup type to automatic
        -d <service dependency> - Can be entered multiple times

jk_nt_service -i tomcat_ils wrapper.properties
The service named tomcat_ils was created. Now adding registry entries
Registry values were added.
If you have already updated wrapper.properties you may start the
tomcat_ils service by executing "jk_nt_service -s tomcat_ils" from the
command prompt
```

The service has been set up and is available in the **Services** dialogue window of the operating system. All further settings can be made there.
Change to the **Services** dialogue window (see Fig. 4.5).
Request the properties of the tomcat_ils service.
Select the **Local system** as the user account and **Automatic** as the launch type.
The service setup is complete. Do not start the service for now.

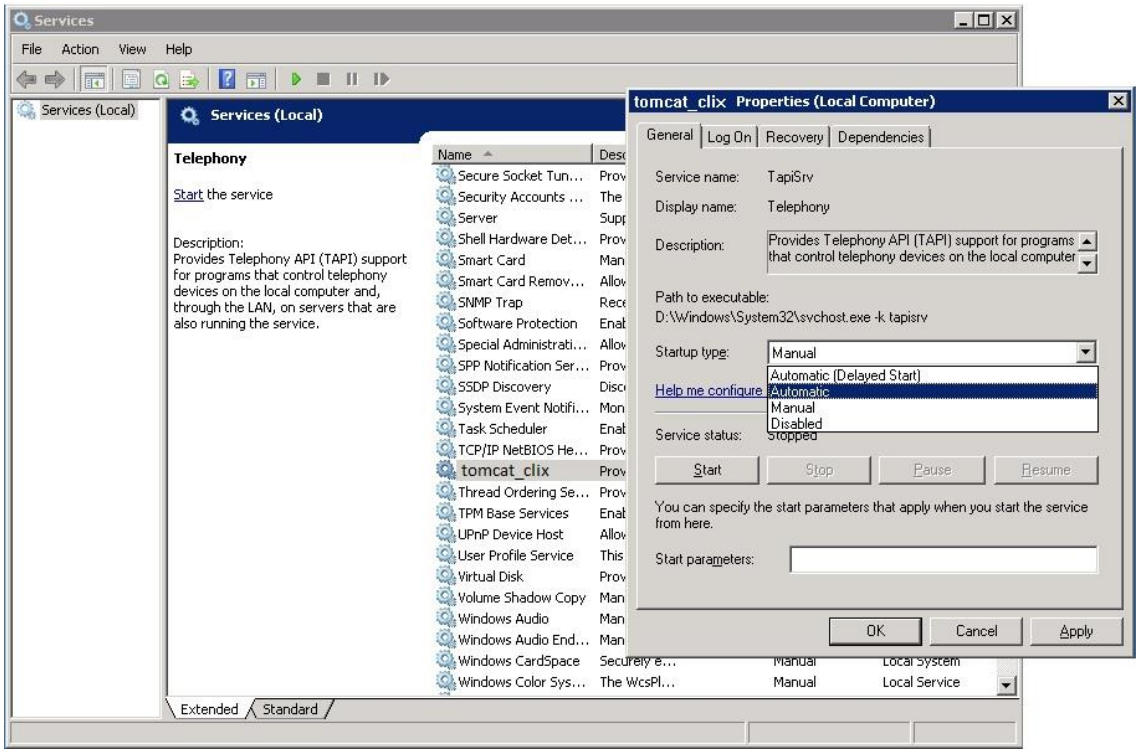

Fig. 4.5: Services dialogue window in Windows

4.12 Test ILS installation

4.12.1 Test availability of the server

The application and database servers must each have static IP addresses. Allocate a DNS name, at least to the application server, in addition to the static IP address. Users request the ILS application using the DNS name.
The following table contains the IP addresses and DNS names used in the examples of these instructions. Replace these example values with the actual values of your installation.

Server	IP address	DNS name
Application server	192.168.0.5	ila.yourdomain.com
Database server	192.168.0.7	dbila.yourdomain.com

It must be possible to contact the application server from a remote computer. It must be possible to contact the database server from the application server. These connections must be quick and resilient for the performance of ILS.

The following test assesses the availability of application and database servers in the IP network.

```
# Ping application server via IP address
ping 192.168.0.5

# Ping application server via DNS address
ping ila.yourdomain.com

# Ping database server via application server
ping 192.168.0.7
```

Alternatively, you can also test the availability of the server with **`telnet`**.

4.12.2 Initial upgrade of the database

In the following section, you will launch the newly set up Tomcat service for the first time. The system will carry out the automatic upgrade if the database is not up to date when the application is launched for the first time. For this purpose, ILS checks the database each time the service is launched and, if the database is not up to date, executes the migration scripts contained in the delivery package. This process can take several minutes and will delay the display of the ILS home page accordingly.
The initial upgrade process will be logged in the file clix.log and should be tracked in all cases. The path to the log file clix.log was established in the file **log4j.properties** (see section 4.4.2). The initial upgrade has been successful if the log file contains the message Database upgrade completed successfully and ends with the message – `ClixInitializer.initialize: exit –`.
If errors occur during the initial upgrade, the message Database upgrade aborted will appear in the log file.

4.12.3 Test ILS application locally

Although ILS is operated within the productive environment on a preset web server, we recommend testing the accuracy of the configuration at the start without a separate web server. Therefore, you should test the ILS application locally on the application server.

!	Description
!	**Do not interrupt initial upgrade!** In the following test, the application will be launched for the first time. The system will carry out the initial automatic upgrade if the database is empty. Do not interrupt the initial upgrade process under any circumstances!

Test ILS application:

1. Start the application by launching the Windows service `tomcat_ils` (see section 4.11.13). You may have given the service another name.
2. The system will begin with the initial upgrade of the database. This process may take several minutes. Do not interrupt the process.
3. Follow the progress of the initial upgrade in the log file clix.log.
4. As soon as the messages "`Database upgrade completed successfully`" and "`ClixInitializer.initialize:exit`" appear in the log data, the initial upgrade is complete. The application test was successful. Now install the web server.
5. If an error message appears ("`Database update aborted`"), take the following steps.

Fix incorrect initial upgrade:

1. Stop the service `tomcat_ils`.
2. Identify the cause of the error in the log file `clix.log`.
3. Remove the cause of the error (e.g. incorrect information in `server.xml`).
4. Delete the database and then recreate the database (do not forget new user allocation!).
5. Re-start the service `tomcat_ils` and follow the steps from the previous instructions "Test ILS application".

4.13 Install Web Server

In a productive environment it may be necessary to run ILS with an upstream web server. In particular if

- the application will be run under a privileged TCP-Port (< 1024).
- the application will be run under SSL (https)
- the ILS add-on components "content protection" will be installed.
- all web access is to be logged in detail.

The task of the web server is to receive client requests and pass them on to the servlet engine for processing. In turn, the servlet engine sends its responses to the web server, which then passes them on to the client. To enable these mechanisms to take place, the web server and servlet engine need to be connected with each other via an ISAPI connector. In this setup, the CLIX application must be called up via the port used by the web server. The default is port 80, and the default port for the http log is:
http://ila.yourdomain.com/ils

Webserver Microsoft Internet Information Services (IIS), which are pre-installed on the Windows servers, is described in this document.

4.13.1 Configure ISAPI connector

A connector enables communication between the web server and servlet engine. The ISAPI connector **`isapi_redirect.dll`** is required for the combination IIS/Tomcat. The connector is available to download from the Apache Tomcat website:
https://archive.apache.org/dist/tomcat/tomcat-connectors/jk/binaries/windows/tomcat-connectors-1.2.40-windows-x86_64-iis.zip
Click on the "Binary reserves" link and download the connector. The current documentation for the Apache Tomcat connectors is available here:
http://tomcat.apache.org/connectors-doc/
http://tomcat.apache.org/connectors-doc/webserver_howto/iis.html
http://tomcat.apache.org/connectors-doc/generic_howto/workers.html

Install isapi_redirect.dll:

1. Download the connector `isapi_redirect.dll` (see above).
2. Make sure that the name of the file `is isapi_redirect.dll`.
3. Create a new directory for the connector. Any location can be selected for the directory, although we recommend that the connector directory be saved in the same directory as Tomcat and JDK, e.g. `C:/appserver/environment/IIS_TC_connector_ila2013`
4. Create the following subdirectories:
   `bin`, `conf` and `logs`.
5. Save the connector in the subdirectory bin, e.g.
   C:/appserver/environment/IIS_TC_connector_ila2013/bin/isapi_redirect.dll.

Adjust isapi_redirect.properties file:

`isapi_redirect.properties` is the connector's configuration file and must be adjusted. Create a file **`isapi_redirect.properties`** and add the paths to the configuration files **`workers.properties`** and **`uriworkermap.properties`** to `isapi_redirect.properties`. You will create these paths in the next step.

1. Create a simple text file with the name isapi_redirect.properties to the directory C:/appserver/environment/IIS_TC_connector_ila2013/bin.
2. Add the following content to the text files. Also adjust the content according to your installation (see sections emphasized in bold).

```
# Configuration fileisapi_redirect.properties for ISAPI connector.
# Path to ISAPI extension.
# Virtual subdirectory of the website with execution rights
extension_uri=/jakarta/isapi_redirect.dll
# Absolute path to log file for the ISAPI redirector
log_file=C:\appserver\environment\IIS_TC_connector_ila2013\logs\is-
api.log
# Protocol level:debug | info | warn | error | trace
log_level=info

# Absolute path to file workers.properties
worker_file=C:\appserver\environment\IIS_TC_con-
nector_ila2013\conf\workers.properties

# Absolute path to file uriworkermap.properties
worker_mount_file=C:\appserver\environment\IIS_TC_con-
nector_ila2013\conf\uriworkermap.properties
```

3. To ensure that the file isapi_redirect.properties is found, also copy the file into the directory C:/appserver/environment/IIS_TC_connector_ila2013/conf.

Adjust workers.properties:

The file **`workers.properties`** describes the connection settings of the Tomcat process to the connector.

1. Create a simple text file with the name workers.properties in the directory C:/appserver/environment/IIS_TC_connector_ila2013/conf.
2. Copy the following minimum content into the file and adjust the content:

```
#List of worker who the connector will work with, here only
#"ajp13_clix2010", name freely selectable
worker.list=ajp13_ila

#Assigns the ajp13 log to the worker "ajp13_ila".
#The name and type do not need to be the same!
worker.ajp13_ila.type=ajp13

#Enter the DNS name or IP address of the application server as the
host, #in this case localhost
worker.ajp13_ila.host=localhost

#Port = connection port which is also defined in the <Connector> ele-
ment in the
#server.xml
worker.ajp13_ila.port=8009

#Define worker status:
#worker.jkstatus.type=status
```

3. Save the file workers.properties.

Adjust uriworkermap.properties:

The file **`uriworkermap.properties`** establishes which browser queries are passed on to the Tomcat connector. Ensure that the connector is referenced with the same identifier as defined in the file **`workers.properties`**. For example, the mapping `/*=ajp13_clix2010` ensures that all queries are passed on to the connector. In normal cases, however, not all browser queries need to be processed by the Tomcat instance. Mapping of the JSP pages and servlet queries are therefore sufficient in many cases (see “standard mapping” in the following example).Further mapping options are summarized in the following example:

1. Create a simple text file with the name uriworkermap.properties in the directory C:/appserver/environment/IIS_TC_connector_ila2013/conf.
2. Copy the following content into the file and adjust the content:

```
# uriworkermap.properties - IIS
# This file provides example mapping for the workers defined in the
# file workers.properties
# General syntax for an entry:
# [URL]=[Worker name]
/*=ajp13_ila

# Example with specific context path
#/%contextPath%/*=ajp13_ila

# Optional filters for all JPEG files in context
# Without mapping, the URL must begin with an exclamation point (!)
  #!/servlets-examples/*.jpeg=ajp13_ila
# integrate jkstatus as /jkmanager
# In production environments, access to the URL /jkmanager
# must be secured
 #/jkmanager=jkstatus
# Standard mapping:
  #/*.jsp=ajp13_ila
  #/servlet/*=ajp13_ila
```

3. Save the file uriworkermap.properties.

4.13.2 Install ISAPI filter

Once all configuration files have been adjusted for the connector, the connector can be installed in the IIS manager. The “default web site” can be used as a web site for the CLIX application in IIS.

Add ISAPI limitation:

1. Open the IIS manager.
2. Goto the WebService where the ISAPI connector should be installed.
3. Within this service double click on **ISAPI and CGI limitations** in the **Features** view.
4. Click on **Add** in the area **Actions**.
5. The dialogue field **Add ISAPI or CGI limitation** will be displayed.
6. Enter the path isapi_redirect.dll in the field **ISAPI or CGI path**. In our example, this would be C:/appserver/environment/IIS_TC_connector_ila2013/bin/isapi_redirect.dll.
7. Enter “jakarta” in the field **Description**.
8. Activate the option **Allow extension path to execute** so that the limitations can be carried out automatically.
9. Click on **OK**.

Add ISAPI filter:

1. Select “Default Web Site”
2. Double click on **ISAPI filter** (see Fig. 4.6 below) in the **Features** view.
3. Click on **Add** on the **ISAPI filter** page in the area **Actions**.
4. The dialogue field **ISAPI filter** will be displayed.
5. Enter “jakarta” in the field **Filter name** (see Fig. 4.7 below).
6. Enter the path to the file isapi_redirect.dll, in our example C:/appserver/environment/IIS_TC_connector_ila2013/bin/isapi_redirect.dll, to the field Executable data.
7. Click on **OK**.

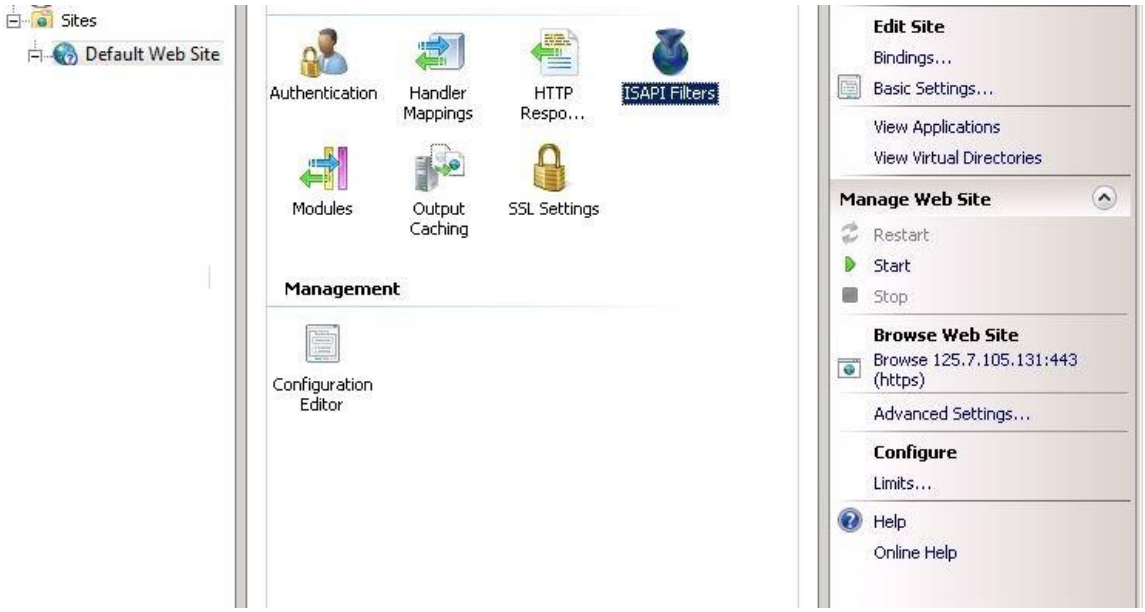

Fig. 4.6: Request menu item ISAPI filter

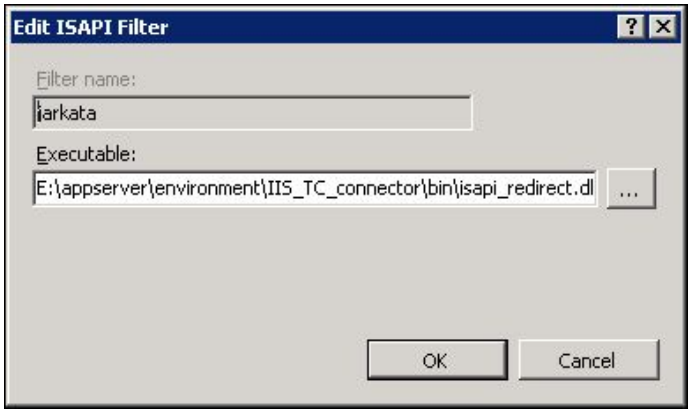

Fig. 4.7: Add ISAPI filter

Handler Mappings

1. Select the Server.
2. Double click on **Handler Mapping** (see Fig. 4.8 below) in the **Features** view.
3. Select **ISAP-dll** and click on **Edit Feature Permissions** in the area **Actions**.
   → The dialogue window will be displayed.
4. Check **Executable** and click on **OK**.

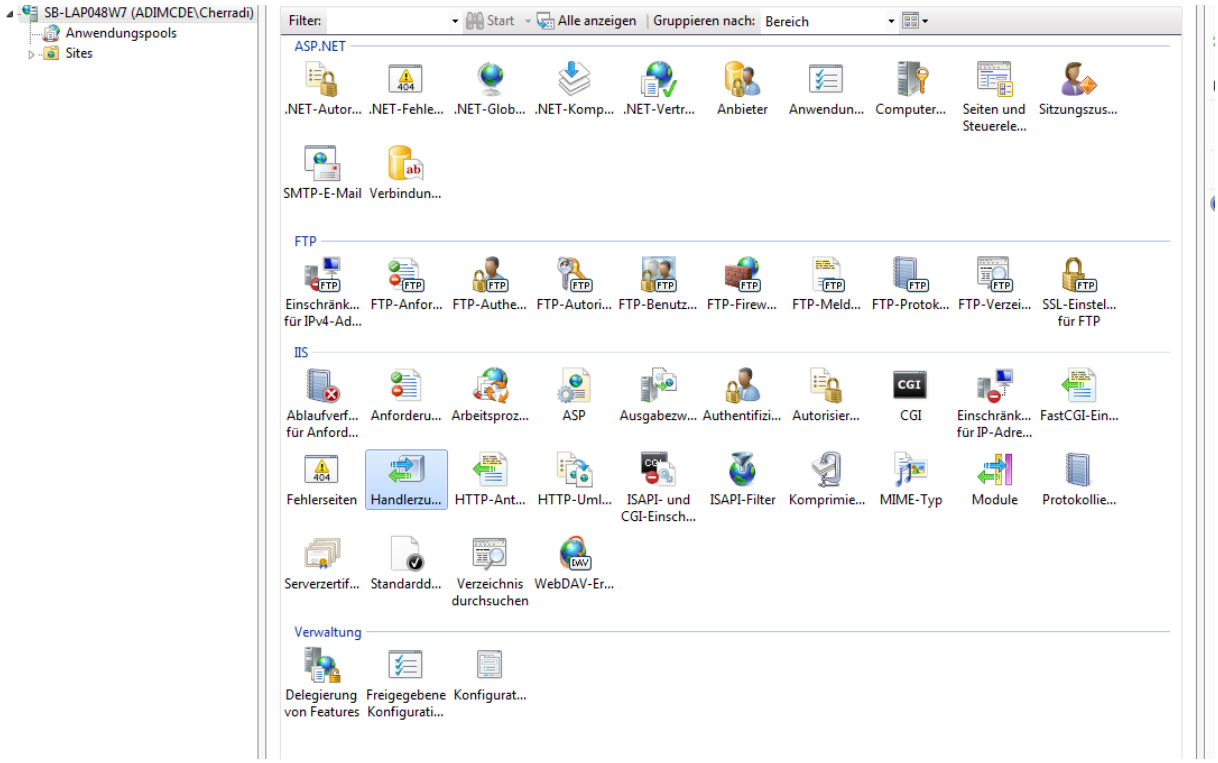

Fig. 4.8: Select Handling Mapper

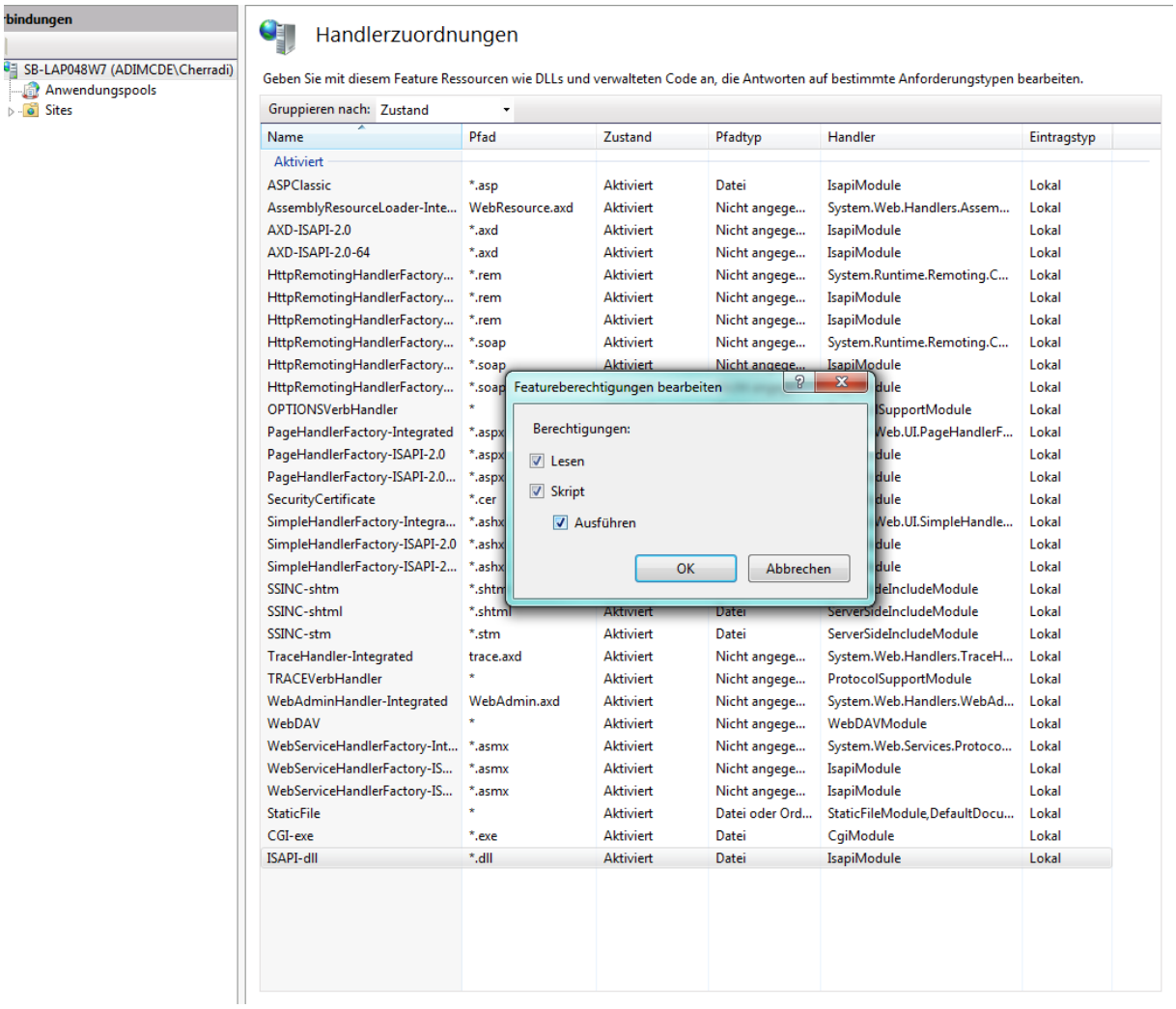

Fig. 4.9: Allow the execution of ISAP

Request filtering

The default Maximum Upload of IIS server is 30000000 bytes, which means Maximum upload of 30MB, it must be increased by executing the following steps.

1. Select the Default Web Site.
2. Double Click on **Request Filtering** in the **Features** View.
3. Click on the **Edit Feature Settings**, a window will be displayed.
4. You have to increase the number of the field **Maximum allowed content length**.
5. Click **OK**.

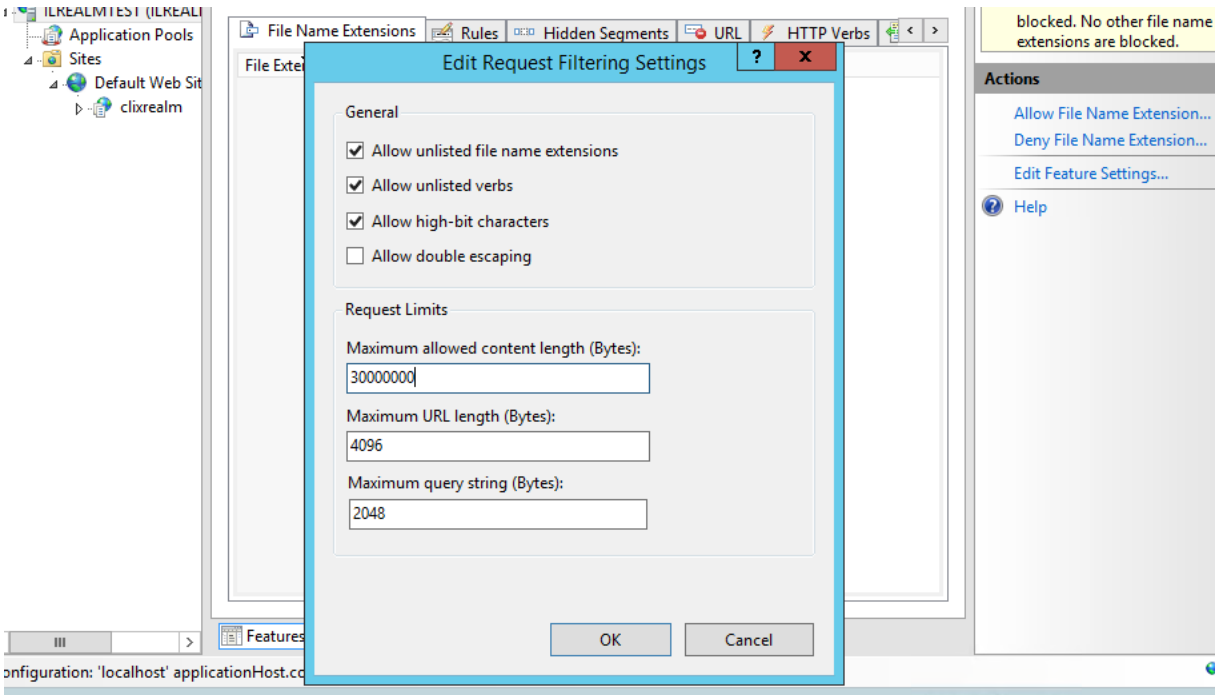

Fig. 4.10: Increase the upload size

4.13.3 Create virtual directory for connector:

1. Select the default website in the IIS manager.
2. Click on View Virtual Directories in the area Actions.
3. Click on **Add Virtual Directories** on the page **Virtual Directories** in the area **Actions**.
   → The dialogue field View Virtual Directory is displayed (see Fig. 4.11 below).
4. Enter “jakarta” in the field **Alias** (must match the name of the virtual directory in the file `isapi_redirect.properties`, see section 4.13.1).
5. Enter the path to the subdirectory in the field **Physical path**, in our example `C:/appserver/environment/IIS_TC_connector_ila2013/bin`.
6. Click on **OK**.
7. The virtual directory **Jakarta** will appear in the IIS Manager below the standard website entry (see Fig. 4.12 below).

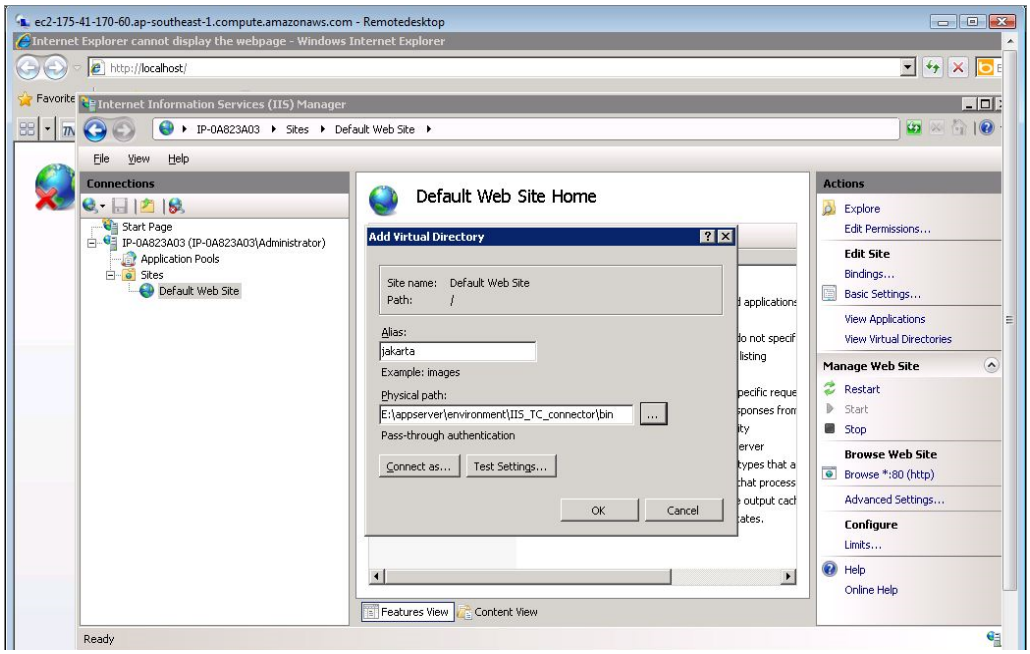

Fig. 4.11: Add virtual directory

1. Right-click on the virtual directory **jakarta** in the IIS Manager. It is displayed under the standard website entry (see Fig. 4.12 below).
2. Select Convert to Application in the context menu.
   → The connection is closed.

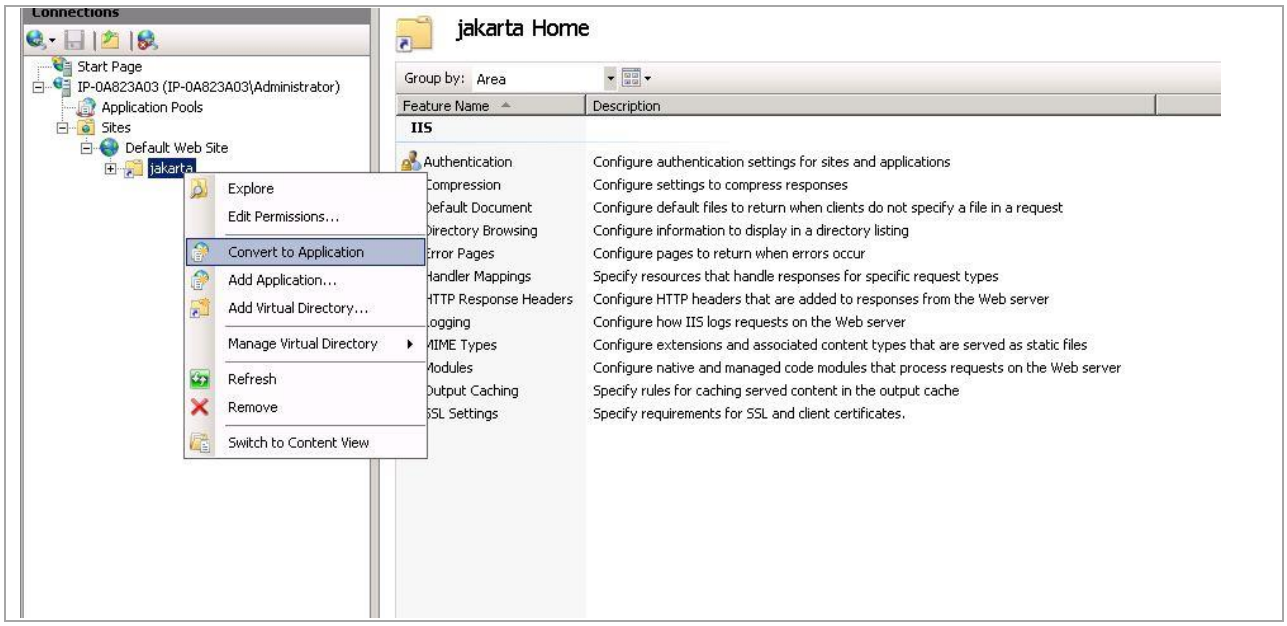

Fig. 4.12: Convert to application

4.13.4 Trouble Shooting

If the IIS, Tomcat and the ISAPI filter have been correctly configured, but calling up CLIX results in a 500 error, check whether the user under which the IIS is being run has read-only and execute rights on the following directory (in our case):
`C:/appserver/environment/IIS_TC_connector_ila2013/`

If the corresponding IIS user is not recognized, the user "Everyone" can be assigned the appropriate rights:

1. Right-click on the "**IIS_TC_connector_ila2013**" directory and go to "**Features**".
2. Switch to the "**Security**" tab.
3. Click "**Edit**", a pop up opens.
4. Click "**Add**", a pop up opens.
5. Write "**Everyone**" in the text field and click on "**Check name**" → The text should now be underlined.
6. Click "**OK**".
7. Ensure that the *read-only* and *execute* rights have been assigned to the appropriate user.

Exit the dialogue using the "**OK**" buttons.

4.14 Test ILS installation

4.14.1 Test ILS installation locally

Start services:

1. Stop the Tomcat service (in this document, tomcat_ils).
2. Delete all Tomcat and IIS log files.
3. Re-launch the Tomcat service.
4. Wait until the message `ClixInitializer.initialize:exit` appears in the log file `clix.log` (see section 4.4.2).
5. Re-launch the IIS by selecting the name of the server in the IIS manager and then clicking on Restart.
6. Carry out the following tests.

Test 1: Launch ILS home page

1. Open the web browser locally on the application server.
2. Request the ILS home page. Enter *http://localhost/* (or the path that you have given to the frontend / ILP) into the browser.
3. The ILS home page will be displayed.
4. Log into ILS as an ILS learner. The login data has been provided by imc.
5. If the personalized home page is displayed, Test 1 has been completed successfully.

Stay logged in as a learner. Continue with Test 2.

Test 2: Check database function

1. Navigate to the personal home page **My area** > **My settings** > **My profile** > **Edit**.
2. Change you own email address.
3. Log out and then back in.
4. If the changed email address is still displayed, the communication with the database is working correctly.

Continue with Test 3.

Test 3: Check access to the ILS data directory data

5. Request the empty file **`dummy.txt`** in the web browser. Enter *http://localhost/ils/data/audio/dummy.txt* into the browser.
6. If the file `dummy.txt` is displayed, ILS has access to the data directory.
7. Access the ILS home page as described in Test 1.
8. Log in as a tutor.
9. The personalized home page is displayed.
10. Navigate to Content administration > Create > Document.
11. Upload an image file (photo) and save the file.
12. The photo should be displayed in the document preview.
13. Log out and then back in.
14. If the photo is still visible, ILS can successfully load content in the data directory.

Continue with Test 4.

Test 4: Check email dispatch

1. Access the ILS home page as described in Test 1.
2. Log in as an administrator.
3. The personalized home page is displayed.
4. Navigate to User administration > Users.
5. Send your self an email by clicking on the email icon.
6. If the email is received correctly, the email dispatch is working.
7. Remain logged in as an administrator.

Continue with Test 5.

Test 5: Check rendering

This test checks whether ILS has access to system functions for the graphic creation (rendering) of images. These functions are part of the operating system in Windows.

1. Navigate as an administrator to Analysis and reports > Evaluation of user access to the learning platform > Logins per month.
2. Select all groups as the filter and the period from one year ago to today.
3. Select Generate report > Diagram.
4. If the diagram is displayed correctly, the rendering works.

Continue with the final Test 6.

Test 6: Check localization of the database

The final test in this series checks the correct localization of the database using the correct handling of national characters.

1. Log into ILS as an administrator using the German interface.
2. Navigate to Content administration > Metatag > Search.
3. Add “All content” and the name "ü" to the search.
4. Click on **Search**.
5. The character “ü” should be displayed correctly.
6. Create a new meta tag by selecting **Create** > **Meta tag with command line** > **German**.
7. Give the meta tag the name “üöaß”.
8. Click on **Save**.
9. If the name of the meta tag is displayed correctly in the title, the database has been localized correctly.
10. Delete the meta tag.
11. The tests are complete. When all tests have been passed, ILS is fully functional.

4.14.2 Accessing the ILS application from a remote computer

Once the tests in the previous section have been completed successfully, the network connection can be tested. For this purpose, the tests are to be repeated from a remote computer using the DNS addressing. Tests 1 to 3 are particularly important.

Check access to the ILS application from a remote computer:

1. Open the web browser on the remote computer.
2. Request the ILS home page. Enter http://ila.yourdomain.com/ in the browser (adjust http port number).
   The ILS home page will be displayed.
3. Log in as an administrator.
   If the personalized home page is displayed, the first part of the test has been completed successfully.
4. Do the tests described in the previous section 4.14.1.
   If the tests are completed successfully, contact from a remote computer works correctly.

5 Special precautionary measures and tips

5.1 Turn off directory browsing

With the help of a web browser, it is possible in principle to navigate through the directory structure of the installation. This involves several security risks and should be prevented in the productive environment, in particular if the ILS installation is released for access from the Internet.
This directory browsing option must be turned off for both the web server and the servlet engine. Details about how this works in individual cases is available in the respective product documentation.

Turn off DB in IIS:

1. Right-click on the IIS manager on the standard website.
2. Select **Properties** in the context menu.
   → The **Properties** dialogue window will be displayed (see Fig. 5.1 below).
3. Select the **HomeDirectory** tab.
4. Deactivate the checkbox **Directory browsing**.
5. Click on **OK**.
   → Directory browsing has been turned off.

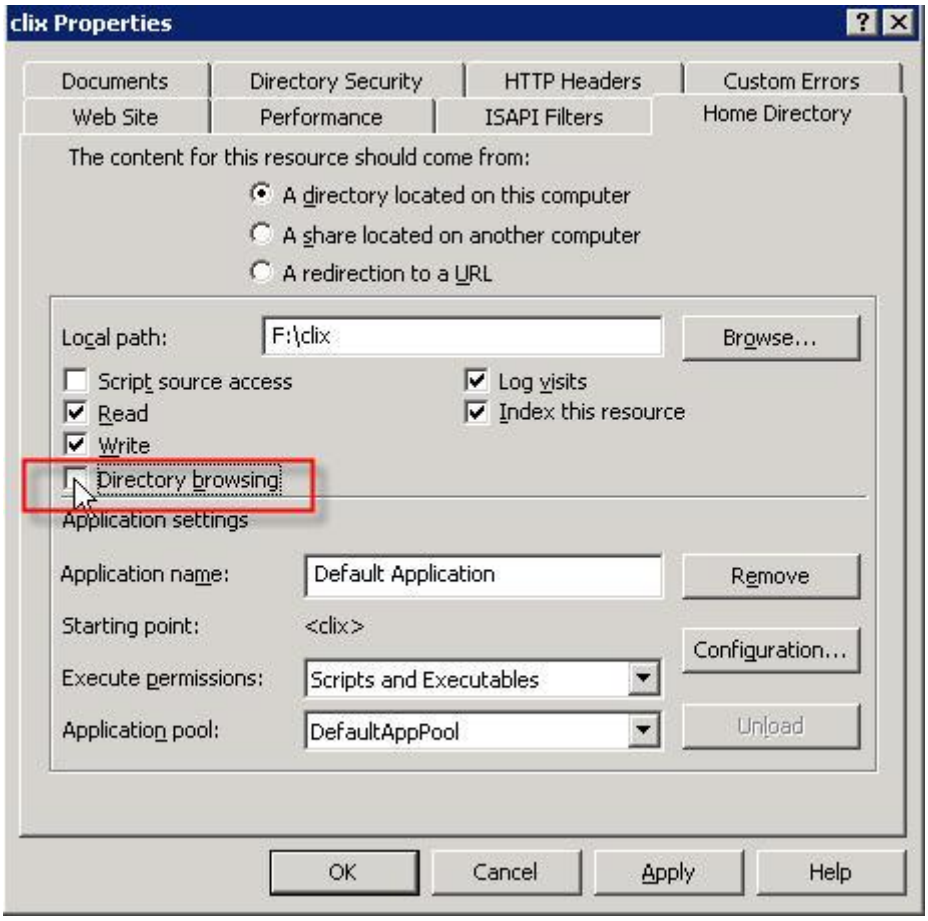

Fig. 5.1: Turn off directory browsing

Turn off DB in Tomcat:

Open the file **web.xml** in the **conf** directory of the Tomcat installation.

1. In the default servlet section, set the parameter listings to false.

```
<servlet>
<servlet-name>default</servlet-name>
<init-param>
<param-name>listings</param-name>
<param-value>false</param-value>
</init-param>
</servlet>
```

2. To check wither the DB is turned off and no subdirectories are visible, access CLIX once via port 80 and once via port 8080:
   http://ila.kunde.de/ils/data/
   http://ila.kunde.de:8080/ilp/data/

5.2 Remove example applications for the servlet engine

Example applications are usually installed during the installation of the servlet engine Tomcat. These applications can be removed.

Remove example applications for the servlet engine:

1. Delete all directories and files from the folder `C:/appserver/environment/tomcat/webapps`.
2. Empty the work directory work of the servlet engine.
   → The example applications have been removed.

5.3 Launch application

The application server and database server may restart after a power cut or another disruption. Launch the required server components in the following order in the event of a restart of this type.

1. Database
2. Servlet engine
3. Web server
4. The components must start in the order stated here for ILS to launch correctly.

5.4 Translate JSP pages

JSP pages are programming files and part of the source code. JSP pages need to be translated before they are executed. If the servlet engine requests JSP pages which have not yet been translated, these JSP pages will first need to be translated. From the point of view of the user, this significantly delays the response rate of the application.
When a freshly installed application is launched for the first time, these delays can be more frequent and lead to timeout exceedances. For this reason we recommend translating all JSP pages after the testing phase.
This occurs either within the scope of an advanced testing phase where all frequently accessed ILS pages are visited. Alternatively, you can use a special JSP compiler.

5.5 Define MIME types

ILS manages various document types as media (e.g. PowerPoint presentations or Excel tables). If such documents are opened in your master application, corresponding MIME types will need to be defined. The required mapping is usually entered into the configuration of the web server.
If you are running the application with no independent web server, there is no MIME support. An alternative is to enter the MIME mapping into the servlet engine configuration.
The MIME mapping in the servlet engine takes place in the file web.xml of the servlet engine (not the ILS application!). The file **web.xml** is located in the subdirectory **conf** of the servlet engine installation directory, e.g. **C:/appserver/environment/tomcat/conf/**.

```
<mime-mapping>
<extension>xls</extension>
<mime-type>application/vnd.ms-excel</mime-type>
</mime-mapping>
<mime-mapping>
<extension>ppt</extension>
<mime-type>application/vnd.ms-powerpoint</mime-type>
</mime-mapping>
<mime-mapping>
<extension>pps</extension>
<mime-type>application/vnd.ms-powerpoint</mime-type>
</mime-mapping>
<mime-mapping>
<extension>properties</extension>
<mime-type>text/plain</mime-type>
</mime-mapping>
```

5.5.1 Enhancements of MIME-Type for „Content Studio“ (HTML5) Contents

For contents which have been produced by the eLearning authoring software “Content Studio” some more Mime-Type definitions are necessary. As described above the mime-type enhancements have to be edited on environment instance which is going to deliver the content. Depending on the connectors configuration the web server and/or on the servlet engine configuration has to be enhanced by the following entries.

5.5.2 Enhancements for IIS Web Server:

In order to add some new mime type definitions in IIS please use administration tool of IIS.

3. Open the IIS management section for the web site you want to enhance the mime types
4. Double-click the MIME Types icon to open the feature.
5. Click Add on the right-hand sidebar
6. Enter the appropriate information: Extension - the file type extension (e.g. .json) and MIME type - the type of file this extension (e.g. application/json).
7. Click OK.

Please repeat the steps above for all the following extensions

MIME-Type	Extension
application/json	json
video/mp4	mp4
audio/mp4	m4a
audio/ogg	ogg
video/webm	webm
image/svg+xml	svg
application/font-woff	woff
application/x-compressed	imc
text/plain	properties
application/xml	xml

5.5.3 Enhancements for Tomcat Servlet Engine:

If the content is delivered directly by the servlet engine the file web.xml placed in the configuration folder of your Tomcat installation (e.g. **…/environment/tomcat/conf/**) has to be enhanced by the following mime mapping entries:

```
<mime-mapping>
        <extension>json</extension>
        <mime-type>application/json</mime-type>
    </mime-mapping>
    <mime-mapping>
        <extension>mp4</extension>
        <mime-type>video/mp4</mime-type>
    </mime-mapping>
    <mime-mapping>
        <extension>m4a</extension>
        <mime-type>audio/mp4</mime-type>
    </mime-mapping>
    <mime-mapping>
        <extension>ogg</extension>
        <mime-type>audio/ogg</mime-type>
    </mime-mapping>
    <mime-mapping>
        <extension>webm</extension>
        <mime-type>video/webm</mime-type>
    </mime-mapping>
    <mime-mapping>
        <extension>svg</extension>
        <mime-type>image/svg+xml</mime-type>
    </mime-mapping>
    <mime-mapping>
        <extension>woff</extension>
        <mime-type>application/font-woff</mime-type>
    </mime-mapping>
    <mime-mapping>
        <extension>imc</extension>
        <mime-type>application/x-compressed</mime-type>
</mime-mapping>
<mime-mapping>
    <extension>properties</extension>
    <mime-type>text/plain</mime-type>
</mime-mapping>
```

5.6 Do not delete pre-defined user accounts

The ILS application is filled with several pre-defined user accounts during delivery. These user accounts are used partially for internal purposes and partially for completing administrative tasks.

!	Description
!	**The following user accounts must not be deleted under any circumstances!** • DUMMYUSER • imc_admin • imc_super • imc_standard • imc_portal • sap_import These user accounts have the internal key values 0, 1, 2, 3, 4, 5 and 6.

!	Description
!	Change passwords! An explicit login for `DUMMYUSER` or `sap_import` is not possible for technical reasons and also not required. Logging into the user accounts `imc_super`, `imc_admin` and `imc_standard` takes place using the passwords of the same name. These passwords must be changed for security reasons!

The user **`imc_standard`** can be used to use the platform with the authorisations of a learner. The user **`imc_admin`**, which is a member of the fully authorised group of administrators, can deal with most administrative tasks in ILS.
Logging in as user **`imc_super`** provides the user with full access to all resources. New clients can be created and new user accounts of the type "`super`" can be created with this user account, as long as this is permitted by the purchased license model.

5.7 Licensing

ILS is delivered with an initial license as default, which provides access to the basic components for several test users.
So that you are able to productively use the platform, adjust the license settings in the application according to your requirements.

Adjust license settings:

1. Log in as user **`imc_super`**.
2. Navigate to menu item System administration>System configuration>Licenses.
3. Create a draft license. Details are available in ILS online help.
4. Download your processed license file.
5. Send the license file to your project contact partner at imc AG as a license query.
6. imc AG will create your license key and inform you about which steps to take next.

6 Annex A

We recommend taking the following measures to improve the performance of the database.

6.1 Integrated Windows authentication

The web server IIS offers the opportunity to protect CLIX from unauthorized access with the help of Windows authentication. The available personal profile information can be used for a single sign-on login (SSO) to CLIX. For this purpose, the following prerequisites need to be fulfilled:

1. The application server must be part of the same domain as the client computers which access CLIX.
2. The website is not accessed via a proxy.

6.1.1 Adjust IIS settings

To be able to use the integrated Windows authentication, the corresponding function needs to be activated in the IIS manager.

Activate integrated Windows authentication:

1. Right-click on the IIS manager on the standard website.
2. Select **Properties** in the context menu.
3. The **Properties dialogue** window will be displayed.
4. Select the Directory **security** tab.
5. Click on Edit in the area Authentication and access management.
6. A new dialogue window will be displayed (see Fig. 6.1 below).
7. Deactivate the checkbox **Enable anonymous access**.
8. Activate the checkbox Integrated Windows authentication.
9. Click on **OK**.
10. Restart the application (in IIS Manager, **Stop** and **Launch**).

Make sure that these security settings are also applied to the virtual directories.

Fig. 6.1: Activate Windows authentication

6.1.2 Adjust Tomcat settings

TOMCAT uses its own authentication process as default: To be able to use the Windows authentication with Tomcat, the configuration file **server.xml** (see section 0) needs to be adjusted as follows:

```
<!-- Define an AJP 1.3 Connector on port 8009 -->
<Connector port="8009" enableLookups="false" tomcatAuthentica-
tion="false" redirectPort="8443" protocol="AJP/1.3" />
```

6.1.3 Adjust ILS settings

Ultimately, ILS needs to be configured so that the SSO authentication is used. For this purpose, the configuration file **businessprocess.xml** needs to be adjusted, which is located in the **conf** directory of the ILS installation:

```
...
<authentication deactivationDays="0" showLanguageSelectBox="false"
showSelfregistrationButton="false" showPasswordRequestButton="false"
showSystemCheckButton="false" requireSecureProtocol="false" localAu-
thenticationModuleReference="LOCAL">
...
<authenticationModuleauthenticationModuleReference="SSO"
profileDataSourceIdentifierRef="LOCAL"/>
<authenticationModuleauthenticationModuleReference="LOCAL"
profileDataSourceIdentifierRef="LOCAL"/>
<authenticationSequenceauthenticationModuleReference="SSO"/>
<authenticationSequenceauthenticationModuleReference="LOCAL"/>

</authentication>
...
<externalSourceDefinition>
...
<SSOremoteUserProfileAttribute="LOGIN"/>
...
</externalSourceDefinition>
...
```

7 Annex B

7.1 Installing IGS/LRS/PRS on existing ILS/ILP

Please follow the Steps specified in the table below to have IGS/LRS/PRS installation on your TOMCAT with existing ILS/ILP installation

Add required Application Configuration	
Prepare IGS instance	Section 0
Prepare LRS instance	Section 4.9
Prepare PRS instance	Section 4.10

Add required Tomee Configuration	
IGS Tomcat configuration	Section 4.11.8
LRS Tomcat configuration	Section 4.11.9
PRS Tomcat configuration	Section 0

7.1.1 Enabling configuration in systemintegration

In the systemintegration.xml file located in '`/instance/ils/`' you can find a section commented as given in the table below, Uncomment the section to have the IGS/LRS/PRS working along with the ILS installation.

```
<remoteApi clientId="ILS">
<remoteApiEntry apiId="EXPERIENCE_TRACK"
oAuthUrl="http://localhost:CHANGE_ME_PORT_TOMEE/igs/oauth/accesstoken"
clientSecret="3df97808-df9f-11e5-b86d-9a79f06e9478"
serverUrl="http://localhost:CHANGE_ME_PORT_TOMEE/igs/restapi"/>
<remoteApiEntry apiId="EXPERIENCE_RULE"
oAuthUrl="http://localhost:CHANGE_ME_PORT_TOMEE/igs/oauth/accesstoken"
clientSecret="3df97808-df9f-11e5-b86d-9a79f06e9478"
serverUrl="http://localhost:CHANGE_ME_PORT_TOMEE/igs/restapi"/>
<remoteApiEntry apiId="XAPI"
oAuthUrl="http://localhost:CHANGE_ME_PORT_TOMEE/igs/oauth/accesstoken"
clientSecret="3df97808-df9f-11e5-b86d-9a79f06e9478"
serverUrl="http://localhost:CHANGE_ME_PORT_TOMEE/igs/xAPI/v1"/>
<remoteApiEntry apiId="PROFILE_API"
oAuthUrl="http://localhost:CHANGE_ME_PORT_TOMEE/igs/oauth/accesstoken"
clientSecret="3df97808-df9f-11e5-b86d-9a79f06e9478"
serverUrl="http://localhost:CHANGE_ME_PORT_TOMEE/igs/profileapi/v1" />
</remoteApi>
```

Adapt the **CHANGE_ME_PORT_TOMEE** placeholder to the corresponding port in which ILS is installed.

7.1.2 Adding Hibernate.jar

To enable Hibernate for Gamification services please add the jar file **hibernate-jpa-2.1-api-1.0.0.Final.jar** to `C:/appserver/environment/tomcat/lib` directory. The jar file could be fetched from **install.zip** -> **environment.zip** -> **environment\tomcat\lib** folder.

7.1.3 Enabling Hibernate for Oracle DB

Also MSSQL is considered default for IGS , if you want to use Oracle DB along with IGS please add the following value to the `wrapper.cmd_line` "`-Dhibernate.dialect= de.imc.igs.core.db. OracleUnicodeDialect`" in `wrapper.properties` file located in `C:/appserver/environment/jk` folder.

8 Annex C

8.1 Enabling protection of data directory

The data directory of ILS/ILP can be protected from unwanted access by using temporary download URLs. To enable this feature add the following line to appserver/instance/ils/systemintegration.xml/contentServer

```
<secureData secretKey="aesSecretKey" initVector="anArbitraryInitV"
urlExpirationSeconds="3600" urlExtendedExpirationSeconds="86400" />
```

8.2 Disabling public access to data directory

Depending on how accessing the data directory on ILS_URL is configured, it is necessary to disable the direct access to protected folders by configuration

8.2.1 web.config on ILS_URL

If the data directory is delivered by IIS it is necessary to disable access to `data/reading|mediafiles| scorm|wbt_aicc|wbt|person|group`) by adding the following rule to `web.config` file at `system.webServer/rewrite/rules`:

```
<rule name="Content Protection" stopProcessing="true">
  <match url="data/(reading|wbt|mediafiles|scorm|wbt_aicc|per-
son|group)/" />
  <action type="CustomResponse" statusCode="403" statusReason="Forbid-
den" statusDescription="Forbidden" />
</rule>
```

8.2.2 data/WEB-INF/web.xml

If data folder is delivered directly by Tomcat it is necessary to disable access by changing `data/WEB-INF/web.xml like this:`

```
<?xml version="1.0" encoding="UTF-8"?>
<web-app>
 <security-constraint>
  <display-name>excluded</display-name>
  <web-resource-collection>
   <web-resource-name>No Access</web-resource-name>
   <url-pattern>/reading/*</url-pattern>
   <url-pattern>/wbt/*</url-pattern>
   <url-pattern>/mediafiles/*</url-pattern>
   <url-pattern>/scorm/*</url-pattern>
   <url-pattern>/wbt_aicc/*</url-pattern>
   <url-pattern>/person/*</url-pattern>
   <url-pattern>/group/*</url-pattern>
   <http-method>DELETE</http-method>
   <http-method>PUT</http-method>
   <http-method>HEAD</http-method>
   <http-method>OPTIONS</http-method>
   <http-method>TRACE</http-method>
   <http-method>GET</http-method>
   <http-method>POST</http-method>
  </web-resource-collection>
  <auth-constraint />
  <user-data-constraint>
   <transport-guarantee>NONE</transport-guarantee>
  </user-data-constraint>
 </security-constraint>
</web-app>
```