ORACLE
INSURANCE

Oracle Health Insurance Back Office

OHIJET Application Installation & Configuration Manual

Version 1.13

Part number: F27899-01

November 26, 2020

ORACLE JET 9.2.0 License

CHANGE HISTORY

Release	Version	Changes
10.18.1.3.0	0.1	• Creation
10.18.1.3.0	0.2	• Revision 07-09-2018 • Add 'Back Office' to 'Prerequisites' • Revised 'Deploy HSL and PSL services'
10.18.1.3.0	0.3	• Revised PSL installation
10.18.1.3.0	0.4	• Revised architecture diagram • Minor textual changes • Revised note about hsl.tokenvalidation.rotor
10.18.1.3.0	1.0	• Reviewed. Slightly adjusted and updated to version 1.0.
10.18.2.0.0	1.1	• Republished with new part nr.
10.18.2.3.0	1.2	• Added description for PSL services deployment. • Added description of delivery of properties file templates.
10.19.1.0.0	1.3	• Added Oracle 6.1.0 JET License text • Renamed to OHIJET Application Installation & Configuration Manual • Revised and updated 'HSL and PSL services'
10.19.1.2.0	1.4	• Updated the list of required PSL services • Updated list of modules which require authorization
10.19.1.3.0	1.5	• Changed location of war files to $OZG_BASE/java • BAC0014J, BAC0016J and BAC0018J added in the list of services which might be deployed and in the list of modules which require authorization
10.19.1.4.0	1.6	• Introduced OHIJET.ear and adapted deployment instructions and references to individual web service deployments.
10.19.2.0.0	1.7	• No changes, republished with a new part number.
10.20.1.0.0	1.8	• Oracle JET version has changed.
10.20.3.0.0	1.9	• Oracle JET version has changed.
10.20.4.0.0	1.10	• Oracle JET version has changed.
10.20.6.0.0	1.11	• Oracle JET version has changed.
10.20.7.0.0	1.12	• Added an extra reference to **Doc[1]** for the deployment of HSL services (HSL_POL. HSL_REL, etc).
10.20.8.0.0	1.13	• Oracle JET version has changed.

RELATED DOCUMENTS

A reference in the text (**doc[x]**) is a reference to another document about a subject that is related to this document.
Below is a list of related documents:

Doc[1] OHI Back Office HTTP Service Layer (HSL) Installation & Configuration Manual (CTA13681)

Contents

1 Introduction

The OHI Back Office JET Application provides web modules for OHI Back Office users.

OHI Back Office web modules are designed from the ground up as productivity tools for power users.

Whereas most of the Forms modules help to navigate through the OHI Back Office data to support multiple potential different processes, the web modules are designed to support a specific process.

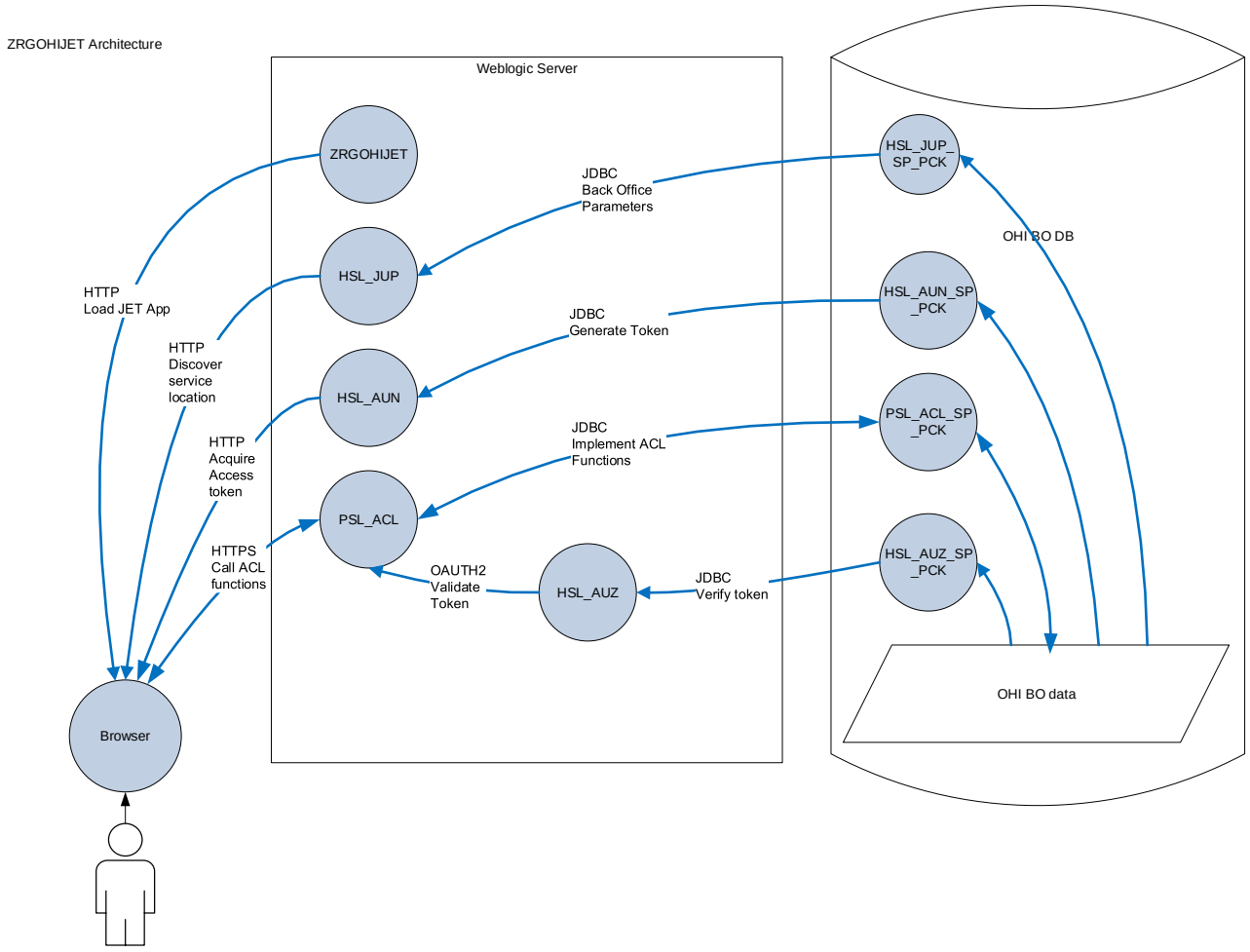
All web modules of the OHIJET application and their supporting components are packaged as a single EAR file: OHIJET.ear.

The OHIJET application is deployed to Oracle WebLogic Application Server (WLS).

This document describes how to install the OHIJET application and its related services.

2 Architectural overview

The diagram below shows the components of the OHI BO JET Application:



2.1 OHIJET Application

The OHIJET application was built using Oracles's JET toolkit for building modern web applications.

Among others, Oracle JET contains UI components, data binding support to synchronize between the front end application and a database back end, and has mobile support.

You can find more information on

http://www.oracle.com/webfolder/technetwork/jet/index.html

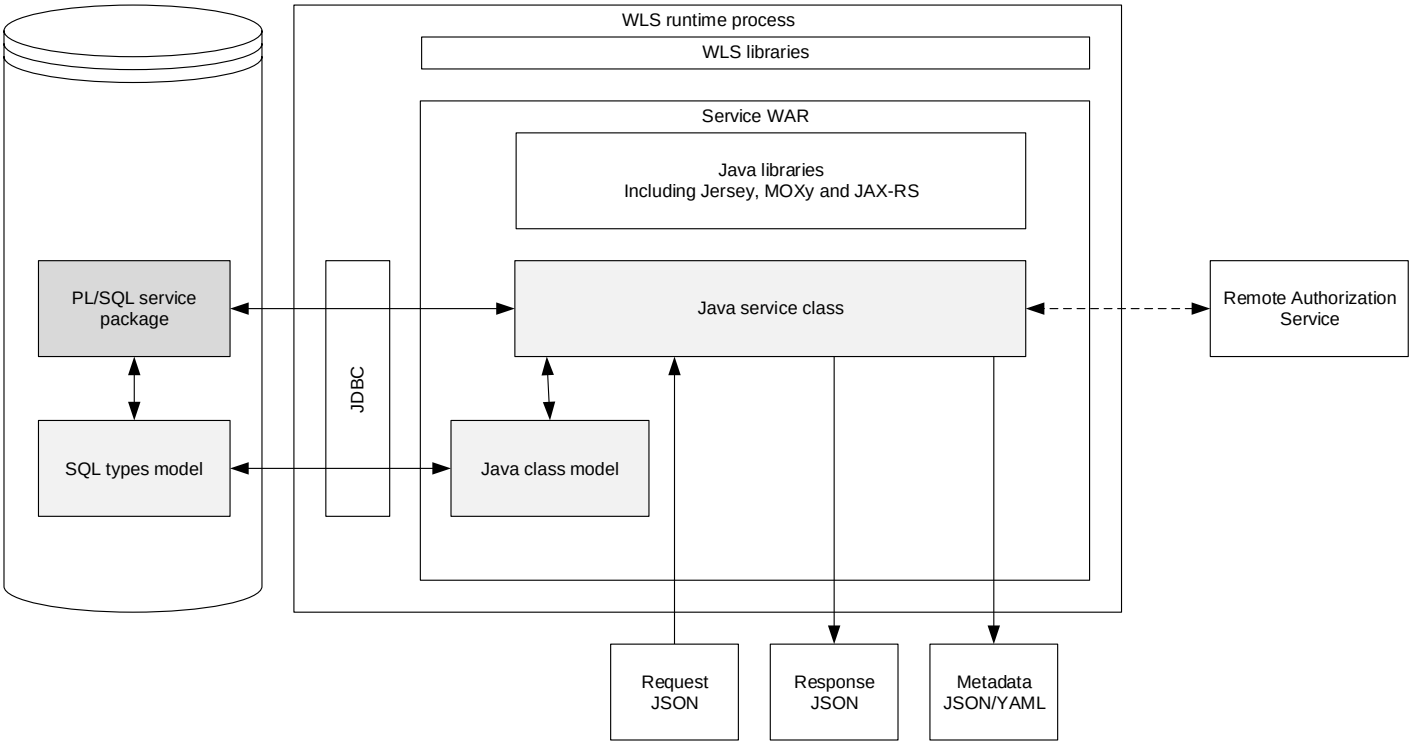
The OHIJET application consists of multiple 'components':

- OHIJET – The JET base web application
- HSL_AUN – Web service to log on to the OHI Back Office database using the credentials entered by the user. If successful, return an OAUTH2 access token and a list of 'claims', i.e. a list of modules which may be accessed by the user.
- HSL_AUZ – Web service to verify that the access token provided by the user is valid for the required service operation.
- HSL_JUP – Web service to retrieve the OHI Back Office parameter values for 'JavaScript user interface' as a list of properties.
- PSL_<app> – Many private REST services used by the JET base web application

The web application calls these HSL and PSL web services to interact with the OHI Back Office database. Ensure that the OHIJET application is deployed successfully. The corresponding EAR file (OHIJET.ear) can be found in $OZG_BASE/java.

2.2 HSL and PSL services

The architecture of each of the HSL and PSL services is shown below:



For more information about HSL services, see **Doc[1]**.

PSL services are created specifically to support OHI BO application components like the OHIJET user interface application. 'PSL' stands for 'Private Service Layer'. These services use the same technology as the HSL services but are not intended as an 'API',

so may and should not be used to support custom client applications as they may be altered without any notice.

Characteristics of PSL services:

- Specifically built to support OHI BO applications components. This means that PSL services are not intended to be called by customer applications. It also means that contents or operation of PSL service operations may be changed by OHI Back Office Development without notice.
- No online help documentation.
- Built on the same technology as HSL services.
- Configured through a 'psl.properties' file, similar to the 'hsl.properties' file used for HSL services.

3 Prerequisites

The following prerequisites apply before you can deploy the OHI Back Office JET Application.

3.1 Weblogic Server (WLS) Preparation

Follow the instructions in **Doc[1]** to prepare WebLogic Server.

Ensure that the following steps are completed:

- Create HSL properties file (as described in **Doc[1]**)
- Create PSL properties file (as described below)
- Add `-Dhsl.properties=<hsl_properties>` to Server Start parameter in WLS. Where <hsl_properties> refers to the location of the HSL properties file (as described in **Doc[1]**).
- Add `-Dpsl.properties=<psl_properties>` to Server Start parameter in WLS. Where <psl_properties> refers to the location of the PSL properties file (as described below).

Note: For OHIJET.ear you may use the same WLS domain as for the HSL and SVL services. At this moment no other 'functional' HSL services are used by the OHI Back Office JET application, so you may deploy other HSL services in the same or a different environment with a different authentication setup.

3.2 Database Preparation

Install the database components for OHI Back Office using OHIPATCH.pl.

3.2.1 Create a HSL database user account

Create a database account to call/use the HSL services, eg. 'hsl_user'.
See **Doc[1]** for more information.

3.2.2 Create a PSL database user account

The PSL services require a PSL database account similar to the HSL database account:

1. Create a database account, for example PSL_USER.
2. Grant create session system privilege to the PSL database account.
3. Log on as the OHI Back Office schema owner, enable serveroutput and run:

```
alg_security_pck.psl_grants
( pi_owner   => '<ohibo_owner>'
, pi_grantee => '<psl_user_account>'
);
```

Example:

```
execute alg_security_pck.psl_grants
( pi_owner   => 'OZG_OWNER'
, pi_grantee => 'PSL_USER');
```

The notes mentioned in **Doc[1]** in the section 'Creating a HSL database account' also apply to the PSL database account.

3.3 OHI Back Office

Do not forget to set up module authorization for the OHIJET modules.

To access module authorization: Systeem > Beheer > Autorisatie > Moduleautorisatie.

3.4 Deploy OHIJET.ear

Like HSL services, the OHIJET.ear file should be deployed through Weblogic Application Server (WLS). The chapter 'Installation of HSL services' in **Doc[1]** also applies to the installation of OHIJET.ear. The instructions below are additional to these general instructions.

The corresponding ear file is available in $OZG_BASE/java.

3.4.1 Create WLS data source for HSL database account

Create a data source for connecting to the HSL user in the OHI Back Office database. Instruction can be found in section 'Creating a data source' in **Doc[1]**.

3.4.2 Create WLS data source for PSL database account

Create a data source for PSL services, similar to 'Creating a data source' for the HSL services (as described in **Doc[1]**). But then refer to the PSL database account instead of the HSL database account.

Also be sure to use the 'psl_grants' packaged procedure instead of the 'hsl_grants' version.

3.4.3 Configure hsl.properties

The properties controlling the HSL_AUN, HSL_AUZ and HSL_JUP services are set in the hsl.properties file. Ensure that these values are set as described in Appendix E & F in **Doc[1]**.

3.4.4 Configure psl.properties

The properties controlling all PSL services are set in the psl.properties file.

With the OHI Back Office release installation, a properties file template called psl.properties.template is distributed to the $OZG_BASE/conf/Back-Office directory. Each OHI Back Office release, may overwrite this template with an updated version. The psl.properties.template can be used as an example to create your own psl.properties file (for example in $OZG_BASE/conf).

Please note that all values are examples. You should consider if these values are appropriate for your installation and replace them with your own values if needed. Values indicated with `<<some_name>>` in the templates are placeholders and must be replaced. This notation is intended to make scripted deployment easier. Also make sure not to set log level to FINE, FINER or FINEST in production mode, use SEVERE or WARNING instead.

The PSL services use the same generic properties as the HSL services (but prefixed with `psl` instead of `hsl`). These generic properties are described in more detail in chapter 'Configuration Files for HSL services' in **Doc[1]**.

The PSL services do support specifying the properties per service, using the <app> abbreviation, but this is not used under normal circumstances. Only when there is a problem OHI Support may ask you to specify for example an enhanced log level and/or separate specific log file name for a specific PSL service for which the abbreviation will be provided at such a moment. The valid abbreviations are not published.

OHI advises you to use the PSL services with OAUTH 2.0 (a 'Bearer' HTTP Authorization header with a JWT token) as authorization method, which is enforced by setting the default `psl.authorization` property to TOKEN. When the `authorization` is set to TOKEN, a number of specific `usercontext` and `tokenvalidation` properties must be set. These properties are explained in more detail in Appendix E in **Doc[1]**, sections 'Setting user context' and 'Access Token Validation'.

3.4.5 WLS Managed Server Start arguments for psl.properties

The instructions for setting the psl.properties parameter are similar to those for setting 'hsl.properties' as described in the 'Installation of HSL services' chapter in **Doc[1]**.

You will need to set `-Dpsl.properties=<filename>`

Example:

```
-Dpsl.properties=/ohi/envBase/vohi/conf/psl.properties
```

Add the line to file $DOMAIN_HOME/bin/setUserOverrides.sh:

```
JAVA_OPTIONS="-Dpsl.properties="/ohi/envBase/vohi/conf/psl.properties"  ${JAVA_OPTIONS}"
```

3.4.6 Deployment

After the weblogic setup is completed and the properties files are configured, the web services can be deployed. Formerly, up to release 10.19.1.3.0, this was done separately. Starting with release 10.19.1.4.0 this is all done with a single application deployment of OHIJET.ear.

The procedure for deploying the OHIJET application is described in chapter 4, Installation of OHIJET Application.

3.4.7 Testing

When you have deployed the OHIJET application, you can test the PSL services by first getting a JWT token through the HSL_AUN service and pass it on a PSL service.

An example call for service PSL_ACL:

```
curl -i -k \

  -H Authorization\:\ Bearer\
eyAiYWxnIjogIkhNQUNfU0gyNTYiLCAidHlwIjogIkpXVCIgfQ.eyAgImV4cCI6
ICIxNTU0OTg3NDg1IiwgImlzcyI6ICJ3d3cub3JhY2xlLmNvbSIsICJwcm4iOiA
iSUxJRUJFUlciLCAibmFtZSI6ICJJbW8gTGllYmVyd2VydGgiLCAiY2xhaW1zIj
ogW3sgIm5hbWUiOiAiWlJHMzA5N0oiLCAicXVlcnlfb25seSI6IGZhbHNlfSx7I
CJuYW1lIjogIkJBQzAwMDhKIiwgInF1ZXJ5X29ubHkiOiBmYWxzZX0seyAibmFt
ZSI6ICJCQUMwMDA2SiIsICJxdWVyeV9vbmx5IjogZmFsc2V9LHsgIm5hbWUiOiA
iQkFDMDAwNUoiLCAicXVlcnlfb25seSI6IGZhbHNlfSx7ICJuYW1lIjogIkJBQz
AwMDdKIiwgInF1ZXJ5X29ubHkiOiBmYWxzZX0seyAibmFtZSI6ICJCQUMwMDA5S
iIsICJxdWVyeV9vbmx5IjogZmFsc2V9LHsgIm5hbWUiOiAiWlJHMzA5OEoiLCAi
cXVlcnlfb25seSI6IGZhbHNlfSx7ICJuYW1lIjogIlpSRzMwOTlKIiwgInF1ZXJ
```

```
5X29ubHkiOiBmYWxzZX1dIH0.SGDDGIV8GdXf-
SFFaMccvm54CClQOExlYd9xExLqxc3Dbkigt9fAEGDmP8jB5xlRw5K1D7b45eij
-QisnJNvFA \

  -H Content-Type\:\ application/json \

  -H Accept\:\ application/json

  -XGET
https\://slc10yip.us.oracle.com\:7410/PSL_ACL/acl/v1/subsequent
check/checks
```

The long string after Bearer is the access token as returned by the HSL_AUN service.

4 Installation of OHIJET Application

The OHI Back Office web modules and REST services are packaged in a single archive named 'OHIJET.ear'. This ear file must be deployed to WLS.

Until OHI Back Office release 10.19.1.3 all warfiles releated to OHIJET were deployed separately. Before deploying OHIJET.ear ensure that none of the following warfiles are deployed:

1. PSL_<app>.war: Ensure that no PSL services are deployed. If PSL services do exist they need to be deleted in the WLS console.

2. HSL_AUN, HSL_AUZ and HSL_JUP: Ensure that none of these HSL services are deployed. If these services do exist they need to be deleted in the WLS console. Note that this does not concern any of the other HSL services (e.g. HSL_POL, HSL_REL, HSL_CLA, HSL_C2B). Follow the instructions in **Doc[1]** for these other HSL services as they are not related to OHIJET.

3. ZRGOHIJET.war: Ensure that this warfile is not deployed. If this war does exist it needs to be deleted in the WLS console.

4.1 OHIJET Deployment

Although the procedure is similar to deploying HSL services you may find it useful to look at the step-by-step instructions below.

1. Select 'Deployments' from the WLS console:

2. Select 'Install' to create a new deployment.

3. Select 'OHIJET.ear'

4. Select to install the deployment as an application

5. Target the application at the managed server on which the OHIJET application should become available.

6. Finish the deployment using 'Custom Roles and Policies'.

The application is now deployed.

4.2 Post Installation

New deployments are not initially active. Either start the new deployment through the WLS console or restart the managed server(s) to which the application is deployed.

4.3 Updating the OHIJET Application

In time, more modules will be added to the OHIJET application.

When deploying a newer version of the OHIJET application, select 'Update' to update the application (ensure that the new OHIJET.ear is copied to the existing location).

Alternatively, select 'Delete' to delete the existing OHIJET application and create a new deployment as described above.

4.4 Deployment validation

When validating the deployment, all components must be correctly installed and deployed.

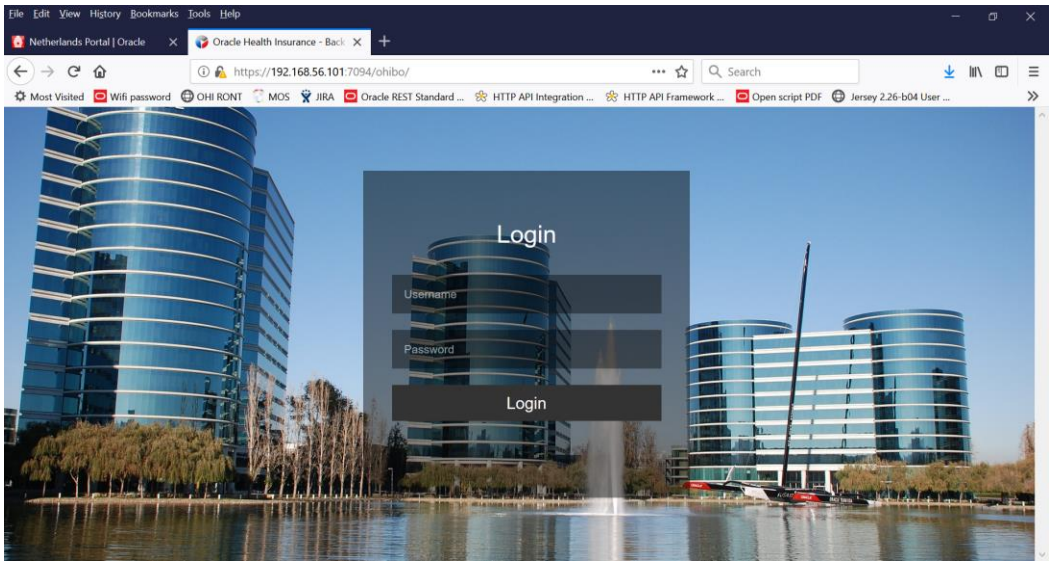
Before you continue:

- verify that you have completed all steps in 'Prerequisites'.
- verify that you have deployed and started the OHIJET.ear application as described in this chapter.

To verify the deployment, browse

https://*server*:*port*/ohibo

Example:



When you have successfully logged on with your OHI usermame and password , the screen should look like this: