



Ihr Partner für IT

# Jasper4Oracle Documentation

Date:	29.11.2018		
Authors:	Philipp Ostmeyer, TEAM GmbH		
Last updated:	26.04.2019	By:	Philipp Ostmeyer

# 1 Index

<b>1</b>	<b>INDEX .....</b>	<b>2</b>
<b>2</b>	<b>NOTES FOR THE READER.....</b>	<b>3</b>
2.1	DOCUMENT HISTORY .....	3
2.2	CONTACT PERSON .....	3
<b>3</b>	<b>INSTALLATION .....</b>	<b>4</b>
3.1	REQUIRED SOFTWARE .....	4
3.2	SERVER COMPONENT INSTALLATION .....	4
3.3	PL/SQL CLIENT-API INSTALLATION .....	5
3.3.1	<i>Grant the necessary authorizations .....</i>	<i>5</i>
3.4	LICENSE KEY INSTALLATION .....	5
<b>4</b>	<b>CONFIGURATION .....</b>	<b>6</b>
4.1	PORT.....	6
4.2	USER INTERFACE.....	6
4.2.1	<i>Admin account .....</i>	<i>6</i>
4.2.2	<i>Data format.....</i>	<i>6</i>
4.3	DEFAULT-DATABASE CONNECTION .....	6
4.4	LOCAL DIRECTORIES.....	7
4.4.1	<i>Library directory.....</i>	<i>7</i>
4.4.2	<i>Deployment directory.....</i>	<i>7</i>
4.5	DRIVER.....	7
4.6	SMTP-SERVER .....	8
4.7	NULL VALUE .....	8
<b>5</b>	<b>USAGE .....</b>	<b>8</b>
5.1	DEPLOYING A REPORT .....	8
5.2	EXPORT OF A DEPLOYED REPORT .....	11
5.2.1	<i>Parameter transfer.....</i>	<i>11</i>
5.2.2	<i>Report export .....</i>	<i>12</i>
5.2.2.1	<i>Parameter .....</i>	<i>12</i>
5.3	PRINTER DISPLAY .....	13
5.4	RELOADING THE REPORT DEFINITIONS .....	13
5.5	MANAGEMENT INTERFACE.....	14
5.5.1	<i>Log-View.....</i>	<i>14</i>
5.5.2	<i>Reloading Reports .....</i>	<i>14</i>
5.6	PL/SQL CLIENT .....	14
5.6.1	<i>View URL .....</i>	<i>16</i>
5.6.2	<i>Download URL.....</i>	<i>16</i>
5.6.3	<i>Print Report.....</i>	<i>17</i>
5.6.4	<i>E-Mail Report .....</i>	<i>18</i>
5.6.5	<i>Saving files locally .....</i>	<i>18</i>
5.6.6	<i>Reloading the report definitions.....</i>	<i>19</i>
<b>6</b>	<b>INSTALLATION VERIFICATION .....</b>	<b>20</b>

## 2 Notes for the reader

### 2.1 Document history

Date	Author	Comment
29.11.2018	Philipp Ostmeyer	Initial Version

### 2.2 Contact person

Name	Accessible via
Philipp Ostmeyer	Tel.: 05254 8008-58 E-Mail: <a href="mailto:po@team-pb.de">po@team-pb.de</a>
Wolf G. Beckmann	Tel.: 05254 8008-39 E-Mail: <a href="mailto:wb@team-pb.de">wb@team-pb.de</a>

## 3 Installation

### 3.1 Required software

Software	Version	Note
Java Runtime Environment	1.8-u152	or higher

### 3.2 Server component installation

Unpack the Jasper4Oracle archive into the installation directory of your choice.  
The installation requires the Oracle JDBC driver (for example „ojdbc8.jar“).  
The driver can either be taken from an Oracle installation ("ORACLE\_HOME / jdbc / lib / ...") or downloaded directly from Oracle.

You can find it under the link:

<https://www.oracle.com/technetwork/database/application-development/jdbc/downloads/index.html>

Copy the "ojdbcXX.jar" file to the resources/jr\_library directory. Depending on the operating system, you can start the server via "startServer" (Linux) or "startServer.bat" (Windows).

Under Linux, the Start script must be made executable. To do this, execute the following command on the command line:

```
chmod u+x startServer  
./startServer
```

```
startServer.bat
```

When calling these commands, functionalities of the OS can be used to redirect the log of the server into files. For example, under Linux:

```
chmod u+x startServer  
./startServer > startServer.log 2&>1
```

### 3.3 PL/SQL Client-API installation

#### 3.3.1 Grant the necessary authorizations

Start SQL-Plus as "sys" database user and grant execute privilege for the packages "utl\_http" and "utl\_url" to the schema, where the TEAM\_J4O package should be installed.

Open a terminal in the database directory of the installation.

```
sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: <SYS_password>
SQL> grant execute on utl_http to <Schema>;
SQL> grant execute on utl_url to <Schema>;
SQL> exit;
```

Install the TEAM\_J4O package and prepare the authorization to connect to the J4O server.

To do this, start SQL-Plus again, but this time connecting to the schema.

Notice that you are in the subdirectory "Database" of the installation directory.

```
sqlplus /nolog
SQL> CONNECT <Schema>
Enter password: <Schema_password>
SQL> @TEAM_J4O.pkh
SQL> @TEAM_J4O.pkb
SQL> @prepareACL
IP or Name of the J4O-Server (eg. localhost): <Server_Name>
Port of the J4O-Server (eg. 8090): <Port_Number>
```

The "prepareACL.sql" script creates a "setup\_acl.sql" file. This is used to grant authorization to connect to the J4O server.

To do this, this script must be run as "sys" user.

```
sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: <SYS_password>
SQL> @setup_acl
```

### 3.4 License key installation

To activate a license key, an additional line in the configuration file ("Jasper4Oracle.config") must be added as follows:

```
LICENSE_KEY=<KEY>
```

## 4 Configuration

### 4.1 Port

To change the port of the Jasper4Oracle server, edit the relevant start script ("startServer.sh" or "startServer.bat").

To do this, you can simply set the "port" parameter to the desired value (e.g. „-port=8090“).

### 4.2 User interface

#### 4.2.1 Admin account

The admin account credentials are defined in the "Jasper4Oracle.config" file. To change these settings, open the file with any editor and change the lines which start with "ADMIN\_USER" and "ADMIN\_PW".

#### 4.2.2 Data format

To change the date format on the J4O server interface, you can change the line beginning with "DISPLAY\_DATE\_FORMAT". All date formats according to Java conventions are permitted here.

A definition of these, you can find here:

<https://docs.oracle.com/javase/8/docs/api/java/text/SimpleDateFormat.html>

### 4.3 Default-Database connection

There is the possibility to set up a default database connection. This is used if no valid database connection for report export is specified.

To set this, the following three lines can be added to the "Jasper4Oracle.config" file:

```
DEFAULT_DB_URL=<DB_URL>
DEFAULT_DB_USER=<DB_SCHEMA>
DEFAULT_DB_PASSWORD=<DB_SCHEMA_PASSWORD>
```

## 4.4 Local directories

### 4.4.1 Library directory

The library directory contains all jar files that are needed at runtime. This includes database drivers, the JasperLibrary as well as the respective archives of the provided reports.

This directory can be configured in the "Jasper4Oracle.config" file. The line beginning with "LIB\_DIRECTORY" contains this definition. By default it points to the "resources" subdirectory of the installation directory.

All paths are seen as subdirectories in the start directory of the J4O server.

If absolute paths are to be used, this definition must be marked with "/" (Linux) or a drive designation, e.g. Start "C: /" (Windows).

The symbol "\" is not allowed and must be replaced by "/".

### 4.4.2 Deployment directory

In the "Jasper4Oracle.config" file, the deployment directory can be configured by changing the beginning line with "DEPLOYMENT\_DIRECTORY".

Locally exported files will be saved relative to this directory. That means, subdirectories will be used/created when needed.

Wenn users are allowed to write to the entire system, not only to a specific directory, use "/" as deployment directory path to grant this access.

Existing files will never be overwritten by the J4O server.

## 4.5 Driver

A driver must be configured for the J4O server. This driver will be used to establish connections during report generation.

In order to use a different driver, its Jarfile must be stored in the "resources/jr\_library" directory. Furthermore, two configuration parameters have to be changed in the "Jasper4Oracle.config" file:

Each JDBC driver is defined by a class and a connector type. These must be specified in the lines "JDBC\_DRIVER\_NAME" and "DB\_CON\_DRIVER\_PREFIX". The driver manufacturer will provide the respective configurations.

E.g.:

```
JDBC_DRIVER_NAME=oracle.jdbc.driver.OracleDriver
DB_CON_DRIVER_PREFIX=jdbc:oracle:thin:@
```

## 4.6 SMTP-Server

The following parameters are required to configure an SMTP server. For each of these parameters there is a distinct line within the configuration file. If they are not available, they can simply be added:

„MAIL\_SMTP\_HOST“ – IP or DNS name of the SMTP server

„MAIL\_SMTP\_PORT“ – SMTP server port

„MAIL\_SMTP\_USER“ – Server authentication user

„MAIL\_SMTP\_PW“ – Corresponding password

If the parameters „MAIL\_SMTP\_USER“ and „MAIL\_SMTP\_PW“ are empty, an attempt is made to establish a connection without authentication.

„MAIL\_SMTP\_FROM\_ADDRESS“ – Default mail address used, if no other is specified

## 4.7 Null Value

Since the communication with the J4O server takes place via http, the parameterization of the reports must be done via strings. To pass an explicit NULL to a report, a NULL string must be configured. To do this, a line with the key "STRING\_PARAMETER\_NULL\_VALUE =" can be added in the configuration file. The desired string can be appended to this line, which will always interpreted as NULL.

If there is no such value configured, there is no way to provide a report with a true "null"-value.

# 5 Usage

## 5.1 Deploying a report

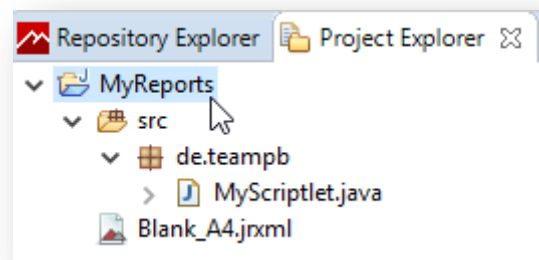
A report is always provided by a "Jar" archive. This contains various components:

JasperReports Files	These files are identified by the extension ".jasper" and are computed by the JasperReports Studio from the "jrxml" files. If a folder structure within a project is used in the JasperReports Studio, this structure must also be retained in the Jar.
Java-Files	These files are identified by the extension "class" and are compiled by the JasperReports Studio from the "java" files. Since Java classes should be in a package, this must be also shown in the folder structure. The JasperReports Studio creates this structure when creating a class or a package. Furthermore, this structure is maintained when generating ".class" files (see "bin"- folder of the project).

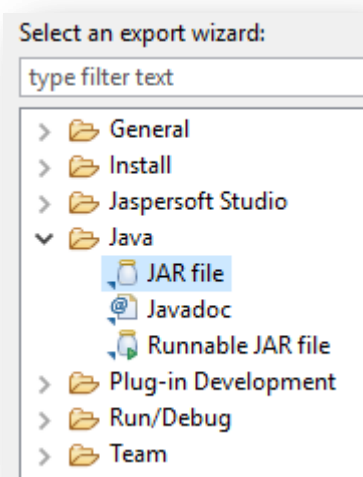
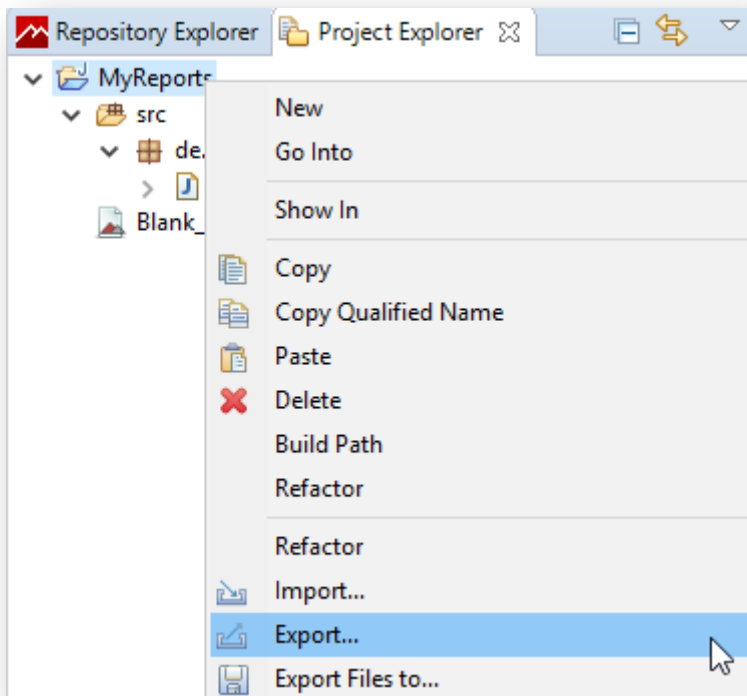


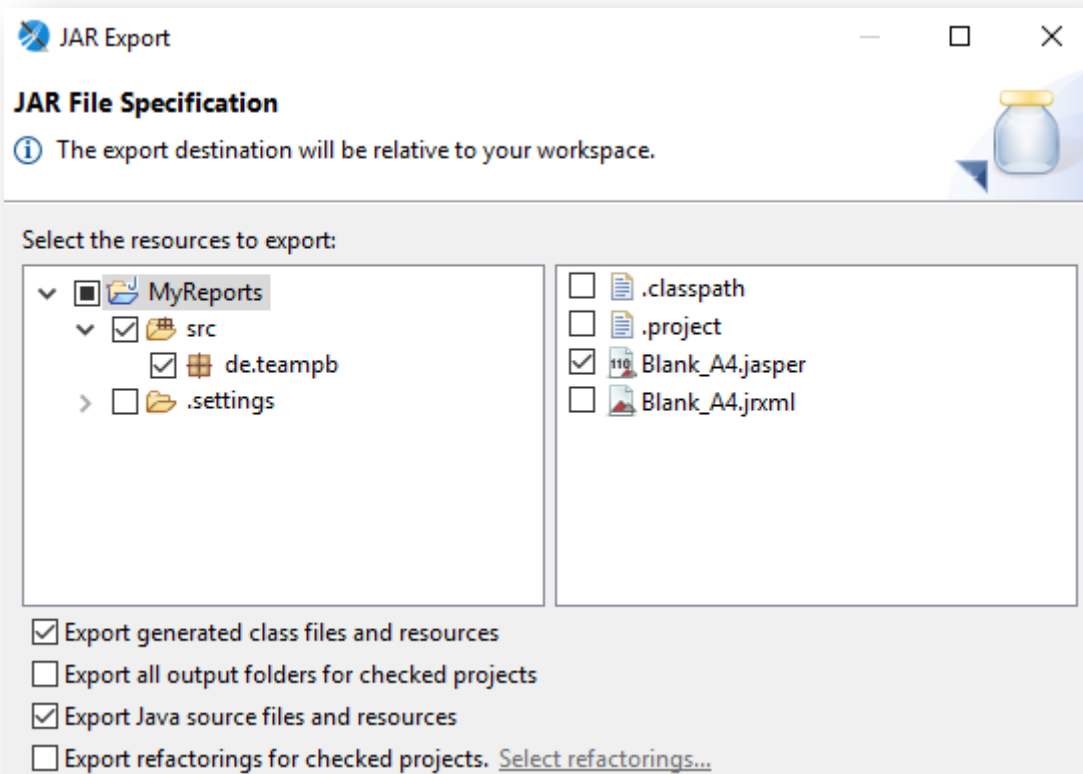
Resource files	Resources are generally defined as all the other files that are required for report generation. These are mainly image files that are to be embedded in the reports. These files should be handled in the same way as JasperReports files. This means that any folder structures in the project should be kept.

A jarfile, like then one shown below, can be built using the JasperReports Studio internal exporter:



1 Right click on the respective projekt





A jar created this way, can then be deployed to the server. It can be placed in one of two directories:

reports	This is the directory in which reports are stored, which are not referenced by other reports. This means that there are no restrictions regarding the reports.
global_libraries	This is the directory in which reports are stored, that are referenced by other reports. The following restrictions apply here: There can not be two ".jasper" files with the same name within all jar archives in this directory. There can not be two ".class" files with the same name within all jar archives in this directory.

Report references are created via the project/source folder reference of the Jasper Studio projects. This procedure enables the developer to use other reports like they are in the same project.

This reference is automatically made on the J4O server and does not require any further configuration.

To activate a new or existing jar, a reload must be started. This can be done via the user interface (see 5.5.2) or via an HTTP interface (see 0).

## 5.2 Export of a deployed report

An export call of a deployed report is divided into the following two http calls:

### 5.2.1 Parameter transfer

The parameters are transferred using an HTTP call. This is defined as follows:

HTTP-Method	POST
URL	<IP>:<PORT>/jasper4oracle/reporting/createToken
Content-Type	„application/x-www-form-urlencoded“
Parameter	Are specified in the body

The parameters of a report are divided into two categories:

Report parameters are used to populate the input parameters defined by JasperReport. These are simply passed in the call.

This means that the names of the parameters do not have to be changed and are passed on directly to the report.

J4O parameters are used to define metaparameters for execution. These includes e.g. report names or database definitions.

The following parameters are defined:

Parameter	Description	Required
j4o_report	Name of the Jar archive to be executed	X
j4o_mainJasper	Name of the „jasper“ file to be executed within the Jar archive	X
j4o_databaseUrl	DNS-Name or IP of the database to which a connection is to be established for the report	(X)
j4o_dbUser	Name of the schema/database user, that will be used to execute the report	(X)
j4o_dbPassword	Database account password	(X)
j4o_format	Generated format. Must include one of the following: PDF, DOCX, HTML, PRINT, CSV	
j4o_locale	The locale used during report execution. The definition can be found at the following link: <a href="https://docs.oracle.com/javase/7/docs/api/java/util/Locale.html#forLanguageTag(java.lang.String)">https://docs.oracle.com/javase/7/docs/api/java/util/Locale.html#forLanguageTag(java.lang.String)</a>	

If the call is successful, a token is returned. The token grants the privilege to export the given report exactly once with these predefined parameters.

### 5.2.2 Report export

For exporting reports, a token generated in **Fehler! Verweisquelle konnte nicht gefunden werden.** is required.

If this token exists, the export process can be started using an HTTP call. This call is defined as follows:

HTTP-Method	GET
URL	<IP>:<PORT>/jasper4oracle/reporting/export
Parameter	Specified in the header of the call

The following parameters are allowed for an export call:

#### 5.2.2.1 Parameter

##### 5.2.2.1.1 „token“

The exported token is passed to this parameter. The parameter is always required.

##### 5.2.2.1.2 „dest\_type“

This parameter describes how the report is made available after the export process. One of the following values is expected:

**VIEW** - The report will be displayed in a browser tab. This value is used if no other value is specified.

**DOWNLOAD** – The report will be downloaded. This allows the user to select a location (if no default folder is set).

**LOCAL\_FILE** - The report is saved on the machine on which the J4O server is started. The parameter "destination" must be set. This is interpreted as a folder specification in which the report is stored. This specification is interpreted relative to the configuration parameter *4.4.2 Deployment* . The parameter "file\_name" must be set. If this is provided with the correct file extension, it is used as defined. Otherwise it will be added.

**PRINT** - The report is sent directly to a printer. The first step is to install the desired printer on the machine on which the J4O server is started. Its name can then be entered in the "destination" parameter, in order to print the report there.  
The parameter "destination" must always be specified if "dest\_type" is set to "PRINT".  
An API endpoint to query the installed printers is provided (see **Fehler! Verweisquelle konnte nicht gefunden werden.**).

#### 5.2.2.1.3 „destination“

This parameter is used by the destination types LOCAL\_FILE and PRINT. More information can be found there.

#### 5.2.2.1.4 „file\_name“

This parameter is used by the destination type LOCAL\_FILE. More information can be found there.

#### 5.2.2.1.5 „print\_copies“

This parameter is used by the destination type PRINT. More information can be found there.

## 5.3 Printer display

The J4O provides an HTTP interface to display the printers available on the machine. This is defined as follows:

HTTP-Method	GET
URL	<IP>:<PORT>/jasper4oracle/admin/printers

This interface is machine-readable.

## 5.4 Reloading the report definitions

In order to activate the provided jar files, these must be recharged. The following HTTP interface can be used for this:

HTTP-Method	GET
URL	<IP>:<PORT>/jasper4oracle/admin/redeploy

The available parameters can be found below:

Parameter	Description	Required
level	Specifies which reports are to be activated. The following values are allowed: „single“ – A single report shall be handled. The parameter „name“ is required for this. „all“ – All files in the „deployment/reports“ directory are handled. „global“ – All files in the „deployment/reports“ and „deployment/global_libraries“ directories are handled.	X
name	Name of the Jar archive to be activated.	(X)

## 5.5 Management interface

The graphical user interface of the J4O server can be accessed via the following HTTP interfaces:


HTTP-Method	GET
URL	<IP>:<PORT>/jasper4oracle/ui/log.xhtml


The credentials are defined in the Jasper4Oracle.config (see 4.2.1).

The interface is optimized for the browser „Google Chrome“, but can also be used with current versions of „Mozilla Firefox“, „Microsoft Internet Explorer“ and „Microsoft Edge“. After logging in, the following services are available:

### 5.5.1 Log-View

The log view shows the status of all current report processes.

The icon  can be used to restrict the status to a certain period of time.

The icon  can be used to open a detailed view. This contains information on parameters and a list of log entries.

### 5.5.2 Reloading Reports

The interface offers the possibility to reload all reports by pressing the button „Redeploy“.

This is equivalent to the call in 0 with the „level“ parameter set to „global“.

## 5.6 PL/SQL Client

The Jasper4Oracle server provides a PL/SQL package that acts as a client interface. Installation instructions can be found under 3.3.

The PL/SQL client serves the HTTP interfaces described above, e.g. it expects the same parameters.

These parameters are divided into the two groups of general and report-specific parameters:

General parameters apply to every call and can be set once per session. This is done via the procedure "team\_j4o.setup\_credentials":

```
procedure setup_credentials(p_username in varchar2 default null,  
                           p_password in varchar2 default null,  
                           p_db_server in varchar2 default null,  
                           p_db_port in varchar2 default null,  
                           p_db_sid in varchar2 default null,  
                           p_db_service in varchar2 default null,  
                           p_db_connect_str in varchar2 default null,  
                           p_j4o_server in varchar2 default null,  
                           p_j4o_port in varchar2 default null);
```

This sets the specified parameters *session*-wide. If this procedure is called several times, the newer parameters overwrites the old ones. „Null“ Parameters are ignored.

It is recommended to call this procedure at the beginning of the session. Especially the J4O server and DB server specific parameters can be stored the same for all.

The following example of such a setup shows that user information can be defined separately from the server-specific information:

```
team_j4o.setup_credentials(p_db_server   => v_db_server,  
                          p_db_port    => v_db_port,  
                          p_db_service => v_db_service,  
                          p_j4o_server => v_j4o_server,  
                          p_j4o_port   => v_j4o_port);  
  
team_j4o.setup_credentials(p_username   => v_db_username,  
                          p_password    => v_db_password);
```

Once these session settings have been made, the report to be executed can be specified:

```
procedure set_report(p_project_name in varchar2,  
                    p_report_name  in varchar2,  
                    p_format       in varchar2 default 'PDF',  
                    p_locale       in varchar2 default 'de-DE'  
);
```

In addition, you can create report-specific parameters. This is done using the following procedure:

```
procedure add_report_param (p_name in varchar2,  
                           p_value in varchar2);
```

This is called once for each parameter of a JasperReport:

```
team_j4o.add_report_param ('departmentName', 'A%');  
team_j4o.add_report_param ('infoText', 'Text');
```

Use the following procedure to delete set parameters:

```
procedure clear_params;
```

Once all parameters have been defined (server, user and report specific), the report can be queried using various functions. All following functions have a return type "Boolean" and at least two out-parameters "p\_error\_code" and "p\_error\_msg".

The return value 0 corresponds to a successful execution and the above mentioned out-parameters do not have to be evaluated.

However, if a 1 is returned, "p\_error\_code" contains the HTTP error code and "p\_error\_msg" contains the error message.

### 5.6.1 View URL

The report can be displayed in any browser using a URL.

A URL is generated with the following function. The generated link can be used exactly once.

```
function get_report_view_url (p_url      out varchar2,  
                             p_err_code out number,  
                             p_err_msg  out varchar2  
    ) return boolean;
```

### 5.6.2 Download URL

The report can be downloaded using any HTTP client via a URL.

A URL is generated with the following function. The generated link can be used exactly once:

```
function get_report_download_url (p_url      out varchar2,  
                                 p_err_code out number,  
                                 p_err_msg  out varchar2  
    ) return boolean;
```



### 5.6.3 Print Report

Furthermore, a report can be sent directly to a printer. You can use the following function to do this:

```
function print_report (p_printer_name in varchar2,  
                      p_copies       in number default 1,  
                      p_err_code     out number,  
                      p_err_msg      out varchar2  
                      ) return boolean;
```

To find out which printers are installed, you can perform the following function:

```
function get_printers (p_printer_list out dbms_utility.lname_array)  
return varchar2;
```

### 5.6.4 E-Mail Report

Furthermore, a report can be sent directly to a configured SMTP server as an e-mail. The following function can be used for this purpose:

```
function mail_report (
    p_from_mail    in varchar2 default null,
    p_to_mail      in varchar2 default null,
    p_cc_mail      in varchar2 default null,
    p_bcc_mail     in varchar2 default null,
    p_subject_mail in varchar2 default null,
    p_text_mail    in varchar2 default null,
    p_report_name  in varchar2 default null,
    p_err_code     out number,
    p_err_msg      out varchar2
) return boolean;
```

Furthermore, there exists an equal function that replaces the type of "p\_text\_mail" with CLOB.

The configuration of the SMTP server can be found in 4.6.

### 5.6.5 Saving files locally

A report can also be saved directly on the server. You can use the following function to do this:

```
function save_report_on_server (
    p_file_name in varchar2,
    p_directory in varchar2 default null,
    p_err_code  out number,
    p_err_msg   out varchar2
) return boolean;
```

The parameter „p\_directory“ is interpreted here relative to the deployment directory. (see 4.4.2)

### 5.6.6 Reloading the report definitions

The process of reloading all reports can be triggered using the following function:

```
function redeploy return number;
```

All reports are reloaded from their respective jar archives. This can be used to release reports for execution.

## 6 Installation verification

To verify that the J4O server or the PL/SQL-API have been correctly installed, the file "TestReportCall.sql" can be found in the "Database" subfolder.

The subfolder contains scripts to showcase different ways to export a report. Copy the first code package into a PL/SQL IDE. After that you have to adjust the credentials to your environment (see 5.6 „setup\_credentials“).

The selected report is available within the standard installation. Running the parameterized script will write an URL to DBMS output. You can open it in any browser.

If you see the TEAM logo and further information at this point, the J4O server has been installed correctly.