pc

priint:comet

THIS DOCUMENT IS DEPRECATED

Please use the provided information with caution.

The information in this document can still help you to understand and configure the priint:comet software. However, the cited third-party software versions in particular may have changed in the meantime. WERK II can no longer promise that all information corresponds to the current situation!

priint:comet 3.4
Installation Guide

Author: Christoph Soergel

Date: 6/28/2016 12:38:00 PM

Version: 1.2, Revision 16542

Document: InstallationGuide.docx

At A Glance

Table of Contents

1 Preface

You are about to install a priint:comet system.

Since there are a number of integration scenarios, the installation process and required components vary. Setting up a standalone InDesign workstation is a quite simple task (using the Installer provided by Werk II), while installing the entire suite, including server components and databases, is rather complex.

This guide will help you to decide which components to install, and provide you with detailed information about installation and configuration of each individual component.

Using the short installation guides at the beginning of the document, you should be able to setup a reliable system step by step with reasonable effort.

1.1 Before You Start

1. **Who are you?**

We assume you are familiar with software installation.

For example, this guide does not cover the installation of a MySQL Database Server (though there are a lot of things to be considered in that context). We expect that you have already installed the Database Server and you have already adjusted the configuration according to the requirements of your application. However, if MySQL requires a certain configuration to interoperate with the priint:comet system (and yes, indeed, it does!), this will be mentioned.

2. **What do you want?**

You don't want to read the whole documentation.

Please first read Section "2 Short Installation Guides" to find out, which setup you need. This section includes short guides, follow these step by step guides and refer to the respective articles for details.

You want to install priint:comet in a server environment.

Though this documentation also includes the workstation setup, you would rather use the *Workstation Installation Guide*, which is also available from Werk II, if you intend to install an InDesign Desktop workstation only.

3. **What is required?**

If you install priint:comet for testing, development or demo purposes, the software will run on virtually any Windows or Mac OS X system. You are welcome to test priint:comet components in any environment; if you encounter problems (or encounter there are no problems) we appreciate your feedback.

Though: **support can only be provided for systems that meet the minimum requirements stated by Werk II**. We highly recommend respecting the requirements of the respective software vendor (e.g. Adobe or Microsoft) and, in addition, considering the scaling requirements available from Werk II.

1.2 Update Existing Installations

Starting with priint:comet Version 3.3, it will be possible to update existing installations.

Older versions may contain customizations, which cannot be covered by general update instructions, so we appreciate to contact support@priint.com in case you intend to update an existing installation.

We will try to find the best solution to get your system up-to-date.

1.3 Get Support

First, the priint.com user forum may be worth a look at: http://forum.priint.com

If you need further installation support, please contact support@priint.com

For license requests (priint:comet Desktop or Server Plug-Ins) please contact license@priint.com

1.4 Typographic Conventions

Paths and names of files, databases, tables, components, applications or menu items appear in *italics*. Very rarely (I hope) you will see important terms set in **bold**, apart from that **bold** type is reserved for headlines and names of chapters or sections.

SQL statements and commands you could enter in a Terminal or command prompt appear in monospace (Courier)

```
read -s -n1 -p "Continue [ Y n ]?" confirmed
```

Output of a commands or fragments of a text file appear in monospace with grey background:

```
# this is a fragment of /etc/my.config.txt
#
# configuration goes here:
```

This is a sequence of installation steps. You are encouraged to follow these instructions strictly step by step.

This is a important note you really should not miss.

These are additional hints not absolutely required to run through the installation. You might find the information useful, but often these sections can be skipped on a first reading.

1.5 Credits

Parts of this document have been adopted from the internal installation wiki, so in one ore the other way nearly everyone at Werk II has contributed to this documentation. Axel Philipsenburg deserves credits for the *InDesignServer Windows Service* Section.

2   Short Installation Guides

2.1   priint:comet / InDesign Desktop

InDesign with priint:comet Plug-Ins.

Usually a datasource is required. This can be either a database server, a SOAP WebService or a XML Offline source.

This setup also includes "Reader installations", if you don't actually need priint:comet functionalities, but only want to process priint:comet documents and preserve comet tags.

2.1	Task	Refer To	Status
Database Server And Databases (if required)			☐
1	Datasource Install datasource (database or WebService) and configure according to priint:comet requirements.	Database, either of • 3.1.1 MySQL • 3.1.2 Microsoft SQL Server • 3.1.3 Oracle WebService • 3.2 SOAP WebService XML Offline • 3.3 XML Offline	☐
2	Configuration and application data	Database • 3.1.4 Database Installation On a Demo / Test machine you will most probably start with a full installation, i.e.: select and install all database components Examples for WebService or XML Offline on request.	☐
InDesign Desktop Workstation			☐
3	InDesign Desktop Install InDesign Desktop and check your system for conformity with ID and priint:comet Plug-Ins	4 Adobe InDesign / priint:comet Plug-Ins	☐
4	priint:comet Desktop Plug-Ins Install and license priint:comet Plug-Ins	4.2 Installing priint:comet Plug-Ins	☐
InDesign Desktop Data Connectivity (if required)			☐
5	Plug-Ins data connectivity Install and configure ODBC drivers or WebService / XML Offline parameters	Database, either of these drivers • 7.1 MySQL / ODBC • 7.2 Microsoft SQL Server / ODBC • 7.3 Oracle / OCI	☐

2.1	Task	Refer To	Status
		WebService	
		• 7.4 SOAP WebService	
		XML Offline	
		• 7.5 XML Offline	
6	Desktop Plug-Ins data connection Configure and test the connection with InDesign Desktop / priint:comet	Database, either of these configurations • 7.1.4 Connect InDesign Desktop With MySQL • 7.2.4 Connect InDesign Desktop With SQL Server • 7.3.4 Connect InDesign Desktop With Oracle WebService • 7.4.2 Connect InDesign Desktop With WebServices XML Offline: • 7.5.2 Connect InDesign Desktop With XML Offline	☐

2.2 priint:comet / InDesignServer

InDesignServer with priint:comet Server Plug-Ins.

This is very similar to a InDesign Desktop installation. InDesignServer can be used for batch processing or controlled via the InDesign Scripting DOM.

This setup also includes "Reader installations", if you don't actually need priint:comet functionalities, but only want to process priint:comet documents and preserve comet tags.

2.2	Task	Refer To	Status
Database Server And Databases (if required)			☐
1	Datasource Install datasource (database or WebService) and configure according to priint:comet requirements.	Database, either of • 3.1.1 MySQL • 3.1.2 Microsoft SQL Server • 3.1.3 Oracle WebService • 3.2 SOAP WebService XML Offline • 3.3 XML Offline	☐
2	Configuration and application data	Database • 3.1.4 Database Installation On a Demo / Test machine you will most probably start with a full installation, i.e.: select and install all database components Examples for WebService or XML Offline on request.	☐
InDesignServer			☐
3	InDesign Server Install InDesignServer and check your system for conformity with IDS and priint:comet Plug-Ins	5 Adobe InDesignServer / priint:comet Server Plug-Ins	☐
4	priint:comet Server Plug-Ins Install and license priint:comet Server Plug-Ins	5.2 Installing priint:comet Server Plug-Ins	☐
5	Configure priint:comet Server Plug-Ins	5.4 InDesignServer / Plug-Ins Command Line Arguments	☐
6	InDesignServer start / stop Consider how to start / stop InDesignServer	5.7 Start / Stop InDesignServer Windows • 5.7.2 Manually Start / Stop InDesignServer On	☐

2.2	Task	Refer To	Status
		Microsoft Windows *or* • *5.7.4* Running InDesignServer As A Windows Service Mac • 5.7.3 Manually Start / Stop InDesignServer On Apple Mac OS X *or* • 5.7.5 Running InDesignServer As A Mac OS X Service using launchd	
InDesignServer Data Connectivity (if required)			☐
7	Plug-Ins data connectivity Install and configure ODBC drivers or WebService / XML Offline parameters	Database, either of these drivers • 7.1 MySQL / ODBC • 7.2 Microsoft SQL Server / ODBC • 7.3 Oracle / OCI WebService • 7.4 SOAP WebService XML Offline • 7.5 XML Offline	☐
8	Server Plug-Ins data connectivity Configure or verify the data connection with InDesignServer and priint:comet Plug-Ins	Database, either of these configurations • 7.1.5 Connect InDesignServer With MySQL • 7.2.5 Connect InDesignServer With SQL Server • 7.3.5 Connect InDesignServer With Oracle WebService • 7.4.3 Connect InDesignServer With WebServices XML Offline • 7.5.3 Connect InDesignServer With XML Offline	☐

2.3 priint:cometserver

This includes several scenarios:

- priint:suite with any of the *Publication Planner*, *Workflow Management* or *Whiteboard* modules
- setups with or without InDesignServer
- priint:cometserver as a simple InDesignServer frontend to control and load balance an arbitrary number of InDesign Servers

The installation process is very similar for these cases and therefore consolidated in the following short guide:

2.3	Task	Refer To	Status
	Database Server And Databases (if required)		☐
1	Datasource Install datasource (database or WebService) and configure according to priint:comet requirements.	Database, either of - 3.1.1 MySQL - 3.1.2 Microsoft SQL Server - 3.1.3 Oracle WebService - 3.2 SOAP WebService XML Offline - 3.3 XML Offline	☐
2	Configuration and application data	Database - 3.1.4 Database Installation On a Demo / Test machine you will most probably start with a full installation, i.e.: select and install all database components Examples for WebService or XML Offline on request.	☐
	InDesignServer (if required)		☐
3	InDesign Server Install InDesignServer and check your system for conformity with IDS and priint:comet Plug-Ins	5 Adobe InDesignServer / priint:comet Server Plug-Ins	☐
4	priint:comet Server Plug-Ins Install and license priint:comet Server Plug-Ins	5.2 Installing priint:comet Server Plug-Ins	☐
5	Configure priint:comet Server Plug-Ins	5.4 InDesignServer / Plug-Ins Command Line Arguments 5.6 Connecting InDesignServer With priint:cometserver	☐
6	InDesignServer start / stop	5.7 Start / Stop InDesignServer	☐

2.3	Task	Refer To	Status
	Consider how to start / stop InDesignServer	Windows • 5.7.2 Manually Start / Stop InDesignServer On Microsoft Windows *or* • *5.7.4* Running InDesignServer As A Windows Service Mac • 5.7.3 Manually Start / Stop InDesignServer On Apple Mac OS X *or* • 5.7.5 Running InDesignServer As A Mac OS X Service using launchd	
GlassFish, priint:cometserver And Whiteboard			☐
7	Java Verify and fix Java installation	• Windows: 6.1.1 Java Installation On Windows • Mac: 6.1.2 Java Installation on Apple Mac OS X	☐
8	GlassFish Installation Install Oracle GlassFish Server 2.1.1	6.2.1 GlassFish Installation	☐
9	GlassFish Configuration Configure the GlassFish application server according to the requirements of priint:cometserver	6.2.2 Required GlassFish Configuration 6.2.3 Recommended JVM Configuration 6.2.4 Recommended HTTP Configuration	☐
10	Libraries and Paths Install and configure additional libraries, support files and paths	6.3.1 Required Libraries, Support Files And Directories	☐
11	JDBC Resources Configure JDBC Resources and Connection Pools	6.3.2 Configure JDBC Resources Includes: Configuration of two Connection Pools: • 7.1.6 Connect priint:cometserver With MySQL • 7.2.6 Connect priint:cometserver With SQL Server • Connect priint:cometserver With Oracle • 7.3.6 Connect priint:cometserver With Oracle	☐
12	Deploy CometServer Deploy the CometServerLight Enterprise Application	6.3.4 Deploy CometServerLight Enterprise Application	☐
13	Whiteboard Install the Whiteboard Flex GUI	6.4 Whiteboard Installation	☐

2.3	Task	Refer To	Status
	GlassFish Integration		☐
14	Configure InDesignServer instances Check, if the default configuration (1 instance) meets your requirements	6.3.5 Configure InDesignServer Instances	☐
15	GlassFish start / stop Consider how to start / stop Oracle GlassFish Server	6.5 Start / Stop GlassFish Server Windows: • 6.5.2 Manually Start / Stop GlassFish On Microsoft Windows *or* • 6.5.4 Running GlassFish As A Windows Service Mac: • 6.5.3 Manually Start / Stop GlassFish On Apple Mac OS X *or* • 6.5.5 Running GlassFish As A Mac OS X Service using launchd	☐
	InDesignServer Data Connectivity (if required)		☐
16	Plug-Ins data connectivity Install and configure ODBC drivers or WebService / XML Offline parameters	Database, either of these drivers • 7.1 MySQL / ODBC • 7.2 Microsoft SQL Server / ODBC • 7.3 Oracle / OCI WebService • 7.4 SOAP WebService XML Offline • 7.5 XML Offline	☐
17	Server Plug-Ins data connectivity Configure or verify the data connection with InDesignServer and priint:comet Plug-Ins	Database, either of these configurations • 7.1.5 Connect InDesignServer With MySQL • 7.2.5 Connect InDesignServer With SQL Server • 7.3.5 Connect InDesignServer With Oracle WebService • 7.4.3 Connect InDesignServer With WebServices XML Offline • 7.5.3 Connect InDesignServer With XML Offline	☐

2.4 priint:comet Developer / Demo / Test machine

Basically this is all of the setups above installed on one machine. If you want to install everything in a convenient order, please follow this installation guide:

2.4	Task	Refer To	Status
Database Server And Databases			☐
1	Datasource Install datasource (database or WebService) and configure according to priint:comet requirements.	Database, either of • 3.1.1 MySQL • 3.1.2 Microsoft SQL Server • 3.1.3 Oracle WebService • 3.2 SOAP WebService XML Offline • 3.3 XML Offline	☐
2	Configuration and application data	Database • 3.1.4 Database Installation On a Demo / Test machine you will most probably start with a full installation, i.e.: select and install all database components Examples for WebService or XML Offline on request.	☐
InDesign Desktop Workstation			☐
3	InDesign Desktop Install InDesign Desktop and check your system for conformity with ID and priint:comet Plug-Ins	4 Adobe InDesign / priint:comet Plug-Ins	☐
4	priint:comet Desktop Plug-Ins Install and license priint:comet Plug-Ins	4.2 Installing priint:comet Plug-Ins	☐
5	Plug-Ins data connectivity Install and configure ODBC drivers or WebService / XML Offline parameters	Database, either of these drivers • 7.1 MySQL / ODBC • 7.2 Microsoft SQL Server / ODBC • 7.3 Oracle / OCI WebService • 7.4 SOAP WebService XML Offline • 7.5 XML Offline	☐
6	Desktop Plug-Ins data connection Configure and test the connection with	Database, either of these configurations • 7.1.4 Connect InDesign Desktop With MySQL • 7.2.4 Connect InDesign Desktop With SQL	☐

2.4	Task	Refer To	Status
	InDesign Desktop / priint:comet	Server • 7.3.4 Connect InDesign Desktop With Oracle WebService • 7.4.2 Connect InDesign Desktop With WebServices XML Offline: • 7.5.2 Connect InDesign Desktop With XML Offline	
InDesignServer			☐
7	InDesign Server Install InDesignServer and check your system for conformity with IDS and priint:comet Plug-Ins	5 Adobe InDesignServer / priint:comet Server Plug-Ins	☐
8	priint:comet Server Plug-Ins Install and license priint:comet Server Plug-Ins	5.2 Installing priint:comet Server Plug-Ins	☐
9	Configure priint:comet Server Plug-Ins	5.4 InDesignServer / Plug-Ins Command Line Arguments 5.6 Connecting InDesignServer With priint:cometserver	☐

10	InDesignServer start / stop
	Consider how to start / stop InDesignServer
		5.7 Start / Stop InDesignServer
		Windows
		• 5.7.2 Manually Start / Stop InDesignServer On Microsoft Windows
		   or
		• *5.7.4* Running InDesignServer As A Windows Service
		Mac
		• 5.7.3 Manually Start / Stop InDesignServer On Apple Mac OS X
		   or
		• 5.7.5 Running InDesignServer As A Mac OS X Service using launchd
			☐

GlassFish, priint:cometserver And Whiteboard			☐
11	Java Verify and fix Java installation	• Windows: 6.1.1 Java Installation On Windows • Mac: 6.1.2 Java Installation on Apple Mac OS X	☐

2.4	Task	Refer To	Status
12	**GlassFish Installation** Install Oracle GlassFish Server 2.1.1	6.2.1 GlassFish Installation	☐
13	**GlassFish Configuration** Configure the GlassFish application server according to the requirements of priint:cometserver	6.2.2 Required GlassFish Configuration 6.2.3 Recommended JVM Configuration 6.2.4 Recommended HTTP Configuration	☐
14	**Libraries and Paths** Install and configure additional libraries, support files and paths	6.3.1 Required Libraries, Support Files And Directories	☐
15	**JDBC Resources** Configure JDBC Resources and Connection Pools	6.3.2 Configure JDBC Resources Includes: Configuration of two Connection Pools: • 7.1.6 Connect priint:cometserver With MySQL • 7.2.6 Connect priint:cometserver With SQL Server • Connect priint:cometserver With Oracle • 7.3.6 Connect priint:cometserver With Oracle	☐
16	**Deploy CometServer** Deploy the CometServerLight Enterprise Application	6.3.4 Deploy CometServerLight Enterprise Application	☐
17	**Whiteboard** Install the Whiteboard Flex GUI	6.4 Whiteboard Installation	☐
Glassfish Integration			☐
18	**Configure InDesignServer instances** Check, if the default configuration (1 instance) meets your requirements	6.3.5 Configure InDesignServer Instances	☐
19	**GlassFish start / stop** Consider how to start / stop Oracle GlassFish Server	6.5 Start / Stop GlassFish Server Windows: • 6.5.2 Manually Start / Stop GlassFish On Microsoft Windows *or* • 6.5.4 Running GlassFish As A Windows Service Mac: • 6.5.3 Manually Start / Stop GlassFish On Apple Mac OS X *or*	☐

2.4	Task	Refer To	Status
		• 6.5.5 Running GlassFish As A Mac OS X Service using launchd	
InDesignServer Data Connectivity (if required)			☐
20	Plug-Ins data connectivity Install and configure ODBC drivers or WebService / XML Offline parameters. If installing on one machine you probably already done this in step (5).	Database, either of these drivers • 7.1 MySQL / ODBC • 7.2 Microsoft SQL Server / ODBC • 7.3 Oracle / OCI WebService • 7.4 SOAP WebService XML Offline • 7.5 XML Offline	☐
21	Server Plug-Ins data connectivity Configure or verify the data connection with InDesignServer and priint:comet Plug-Ins	Database, either of these configurations • 7.1.5 Connect InDesignServer With MySQL • 7.2.5 Connect InDesignServer With SQL Server • 7.3.5 Connect InDesignServer With Oracle WebService • 7.4.3 Connect InDesignServer With WebServices XML Offline • 7.5.3 Connect InDesignServer With XML Offline	☐
Showcase			☐
22	Showcase Installation Additional installation steps for the Showcase	D Showcase Installation	☐

3 Data Sources

The priint:comet system uses data connection for various purposes. There are three main types of data connections

- Database (RDBMS)
- WebService / SOAP
- XML Offline

Which of these connection types is applicable for you, depends on the intended setup and your system environment. When installing priint:cometserver you always need a connection to a RDBMS database and at least the *comet_admin* database schema and tables.

3.1 Database (RDBMS)

Overall there are three database schemas:

- *comet_config*; configuration (such as placeholders, actions, templates) for priint:comet Desktop and Server Plug-Ins
- *comet_data*: business data, such as publications, products, workflows
- *comet_admin*: configuration for priint:cometserver

These schemas can be installed in one, two or three physical databases. Not all of them (and some only partial) may be required for your intended setup.

We provide a set of SQL files and a batch / Unix shell script for database installation. The script is called *InstallDB.bat* resp. *InstallDB.sh* and can be found in the Resources/Install-DB.zip archive (see Section 3.1.4). Use this script to set up installation parameters and choose the components suitable for your needs.

Before importing, make sure your database server meets the requirements to run with the priint:comet system, i.e.: one of the RDBMS systems installed and configured as stated below.

It is not necessary nor recommended installing the components on individual InDesign or Whiteboard workstations (except for test, demo and development system), typically you will install priint:comet database components on a dedicated database server.

3.1.1 MySQL

The recommended and currently supported MySQL Version is

- MySQL Community Server 5.1.63

    http://dev.mysql.com/downloads/mysql/5.1.html/

The following versions have been found to work too, but Werk II does not currently support them (so you will use them at your own risk):

- MySQL Community Server 5.5.24 / MySQL Database 5.5.23 (Enterprise)

    http://dev.mysql.com/downloads/mysql/5.5.html

- MySQL Cluster 7.2.6 / MySQL Cluster 7.2.5 (CGE)

    http://dev.mysql.com/downloads/cluster/7.2.html

Choose the installation package suitable for your platform and OS.

After installation you have to verify and fix the following settings. You can use an administration tool (like the MySQL Workbench, which can be downloaded from http://www.mysql.com/downloads/workbench/), personally I prefer to edit the configuration files (see the hints at the end of the chapter if unsure, where to find the MySQL configuration files):

1. **InnoDB**

   priint:comet requires the InnoDB storage engine. Make sure that InnoDB is neither disabled in the MySQL startup parameters nor in the configuration file (*my.ini* resp. *my.cnf*)

   ```
   # File: my.ini / my.cnf
   [mysqld]
   # Remove or comment out skip-innodb lines in your my.cnf / my.ini
   # file to enable the InnoDB storage engine
   # skip-innodb
   # skip_innodb (MySQL < 4.1, probably still supported in 5.x)
   ```

2. **max_allowed_packet**

   To enable the import of large BLOBs (e.g. InDesign templates), you must increase the value for max packet size. "64M" is a reasonable value, if you work with very large templates, you must set this value even higher.

   ```
   # File: my.ini / my.cnf
   #
   # Add or edit the max_allowed_packet setting in both mysqld and mysql
   # sections
   [mysqld]
   max_allowed_packet=64M

   [mysql]
   max_allowed_packet=64M
   ```

3. **Character set / encoding**

   We recommend setting the default character set for the database server, collation and connection to UTF8. Refer to the MySQL documentation for further information:

   http://dev.mysql.com/doc/refman/5.1/en/charset.html

   A convenient setting would be

   ```
   # File: my.ini / my.cnf
   #
   [mysqld]
   default-character-set=utf8
   # use utf8_unicode_ci for more accurate collation
   default-collation=utf8_general_ci

   character-set-server=utf8
   # use utf8_unicode_ci for more accurate collation
   collation-server=utf8_general_ci
   ```

   If you use the database installation script and let the script create the databases, you don't have to bother about character set and encoding, the script will create the database with correct settings.

4. **Add mysql to your path environment**

   The InstallDB script requires the mysql client program.

   Windows: launch System Properties > Advanced and click "Environment Variables", append the path to the MySQL executables to the Path variable (if not already included)

   Mac: the script will search for the MySQL binaries in the following locations:

/usr/local/mysql/bin/;/usr/bin/;/usr/local/bin/;~/bin/;/bin/

Make sure, the binaries are available in either of these directories. Also you might want to add the MySQL binaries to your PATH variable. You can do so by editing (or creating) the file *~/.bash_profile* and add the following line:

```
# File: ~/.bash_profile
#
export PATH="$PATH:/usr/local/mysql/bin"
```

5. **Set root / admin user password**

   The InstallDB script requires an account with full privileges and non-empty password, so we recommend setting a password for the root account (if not already done). If you run the installation script on the database host, you can limit access to localhost.

After changing any of the configuration settings, you have to restart the MySQL Server. Make sure your settings have been applied, for example, to verify the max_allowed_packet setting, you can launch a SQL Editor (e.g. MySQL Workbench) or the mysql command line client and run the following SQL statement:

```
show variables like 'max_allowed_packet';
```

If you set max_allowed_packet to 64M, the output should show something like

```
+--------------------+----------+
| Variable name      | Value    |
+--------------------+----------+
| max_allowed_packet | 67108864 |
+--------------------+----------+
```

Where can I find the MySQL configuration file?

On Windows systems the file is named my.ini, on Mac or Unix systems my.cnf. This file is searched in several locations upon server startup; please refer to the MySQL documentation for detailed information. A common location is the MySQL Installation directory (e.g. *C:\Programs\MySQL\MySQL Server 5.1*) on Windows or */etc/* on Mac.

If no configuration file exists, you can use one of the templates provided by MySQL (Windows: in the installation directory, Mac: in the support-files folder, usually */usr/local/mysql/support-files*) and adapt it for your needs.

3.1.2 Microsoft SQL Server

The recommended and currently supported Microsoft SQL Server Version is

- SQL Server 2008 R2

After installation verify and fix the following settings:

1. **Enable TCP connections**
   Launch the *SQL Server Configuration Manager*, navigate to *SQL Server Network configuration*, select *Protocols* and make sure, that TCP/IP is enabled.

2. **DB installation user**
   The InstallDB script requires a SQL Server user account with full privileges and non-empty password. The script cannot connect to the database using Windows authentication.

3. **Add the SQL Server Tools directory to your path environment**
   Launch *System Properties > Advanced* and click *Environment Variables*, add the path to the Tools directory (usually something like *C:\Program Files\Microsoft SQL Server\90\Tools\Binn*) to the Path variable (if not already included).

Though you don't have to bother about this anymore (at least when using the SQL scripts provided by Werk II) I think it's worth remembering:

- SQL Server 2005 and earlier: use nvarchar(max) for clobs (such as the actions.statement or panelstatements.statement column)
- SQL Server 2008 and later: use ntext for clobs (such as the actions.statement or panelstatements.statement column)

3.1.3 Oracle

There are lots of things to consider when setting up an Oracle Database server. Using the database installation scripts with the "create databases" option will just create tablespaces, users etc. for *development* and *testing* purposes.

Please refer to your database administrator for settings appropriate for productive environments.

Typically you will install an Oracle 10g or 11g database server on a Unix or Windows server. InDesign Desktop and InDesignServer with priint:comet Plug-Ins connect to the database through the Oracle instant client interface using TNS names for connections, so you will have to enable the TNS listener on the database server.

If you want to use Oracle client tools on Mac OSX, follow these steps:

SQL*Plus On Mac OSX

1. download the **32bit** instant client sqlplus package from the Oracle download site. At the time of this writing, the file is called *instantclient-sqlplus-10.2.0.4.0-macosx-x86.zip*

2. unzip the file

3. do *not* bother the Oracle installation instructions, instead

4. run the following commands in a terminal window

   ```
   sudo mv ~/Downloads/instantclient_10_2-sqlplus32/*.dylib /usr/lib
   sudo mv ~/Downloads/instantclient_10_2-sqlplus32/sqlplus /usr/bin
   ```

5. after that you can run *sqlplus* from any terminal

SQLDeveloper On Mac OSX

1. download SQLDeveloper from the Oracle download site. There's a Mac OSX version available, naturally you will download this version

2. do bother the Oracle installation instructions

3. when connecting to a database, use hostname / IP and port. TNS resolution doesn't work at least on Mac OS X 10.8

Oracle Environment On Mac OSX

In former versions of Mac OSX (including 10.7) it was possible to set environment variables in the *environment.plist* file located in *~/.MacOSX.*

This doesn't work in 10.8 anymore; therefore we recommend the following procedure, which will work on any Mac OSX version:

1. open the file /etc/launchd.conf with superuser privileges in a text editor, e.g.

   ```
   sudo vim /etc/launchd.conf
   ```

2. add the following lines and save the file:

   ```
    # It must be AL32UTF8, but you may want to choose another locale:
   setenv NLS_LANG GERMAN_GERMANY.AL32UTF8
    # This would be a proper place, though you can keep the
    # tnsnames.ora file anywhere else:
   setenv TNS_ADMIN /Library/Preferences/Oracle/
   ```

3.1.4 Database Installation

Use the database installation script provided by Werk II, which can be found in *Resources/Install-DB.zip*.

This script requires a suitable command line client for the respective database vendor (such as the *mysql* tool for a MySQL database or *sqlcmd* for Microsoft SQL Server), therefore we recommend to execute the database installation scripts on the target machine (i.e. the database server host), this will provide best stability and save you further trouble with installing client tools on another machine.

If you choose to create new databases during installation, existing databases with the same name will be dropped and recreated.

If you choose to import in existing databases, you must make sure, that none of the target tables already exists in the database. See Appendix "**Fehler! Verweisquelle konnte nicht gefunden werden. Fehler! Verweisquelle konnte nicht gefunden werden.**" for a detailed list of databases, tables and data imported with each component.

To run the script

1. read the Sections "3.1.5 Database users / accounts" and "3.1.6 Database Components" thoroughly

2. unpack the ZIP archive *Resources/Install-DB.zip* in a directory on a **local drive** (no network mount or virtual disk!)

3. **Windows:** double click InstallDB.bat

4. **Mac:** execute InstallDB.sh from a Terminal (you might have to "chmod +x" the script file before)

5. read and follow the instructions on the screen. In short, you will be asked

   ▪ for the target database (vendor, host, credentials)

   ▪ whether to import in existing databases or create new ones

   ▪ to select from a list of available database modules, see Section "3.1.6 Database Components" for details

   ▪ whether to actually do the import or just preprocess the SQL scripts according to your settings and module selection

The latter will give you more precise control of what is happening to your database server. You will then find a set of SQL files in a folder named "install-tmp", as well as a pretty straightforward Windows batch or Unix shell script, which can be used to import these files on your database server.

3.1.5 Database users / accounts

3.1.5.1 General Information

The priint:comet system uses several authentication layers. Some components require *database authentication*, some use an internal user and access management.

1. **Installation**

   A full privileged *database account* is required to install database components

2. **Login or Connecting with priint:comet Desktop or Server Plug-Ins**

   priint:comet Desktop and Server Plug-Ins use *database authentication* to connect with the *comet_config* database. The associated accounts must be granted with full SELECT / INSERT /

UPDATE / DELETE privileges on all tables. In addition, an entry in the following tables must exist for each account destined for Plug-In login or connection

- *comet_config.iuser*
- *comet_config.userxdomain*

3. **Connecting with priint:cometserver**

   priint:cometserver requires one *database account* to connect to the *comet_admin* and *comet_config* database. This account must be granted with full SELECT / INSERT / UPDATE / DELETE privileges on both databases / all tables.

4. **Login with priint:suite / Whiteboard**
   An entry (but not a dedicated database account) in the following tables must exist for each user destined for priint:cometserver or Whiteboard login

   - *comet_admin.serveruser*
   - *comet_admin.userxrole*
   - *comet_data.wfm_person*

   Usually these accounts are set up using the priint:suite user administration.

5. **Login with both priint:suite / Whiteboard and Plug-Ins**

   This is the typical InDesign Desktop user in a priint:cometserver environment. Though the user does not necessarily use the Whiteboard directly, an account may be required for background operations (like query the status of a publication document from priint:cometserver).
   Therefore you must

   - add a Plug-Ins user as described in (2.)
   - add a priint:suite / Whiteboard user as described in (4.)

   ! The credentials for **database authentication** (for Plug-Ins login) **must match** the **priint:suite / Whiteboard credentials**.

3.1.5.2 Side Effects When Changing Accounts

When changing certain accounts (like the account used for InDesignServer Plug-Ins connection), you also have to change the connection configuration file for InDesignServer. See the sections related to InDesignServer connection (e.g. Section "7.1.5 Connect InDesignServer With MySQL") for details.

When changing an account using the priint:suite user administration, you also have to change the database credentials for this user (unless the account is used for priint:suite / Whiteboard login **only**).

3.1.5.3 Default Accounts

The installation scripts create a couple of default accounts with typical privileges or roles. See section "3.1.7 Standard Accounts" for a list of accounts created during installation.

! If you choose to import the priint:comet database schemas into an **existing** database, you will have to create the *database* accounts manually.

Though it is highly recommended not using the default accounts or at least changing the passwords, keep in mind that this can have quite an impact on other configuration settings.
In order to get the system up easily, I would suggest doing the initial setup with default credentials and changing these settings later.

3.1.6 Database Components

To allow best integration in your environment and potentially existing databases, you can choose from several database installation components.

Typical setups are

1. **Demo / Test**

    Install everything

2. **priint:cometserver / Whiteboard, PIM data in comet standard DB**

    Install everything but *Product Data*, if no InDesignServer is installed also omit *InDesignServer Configuration*.

    Next step will be to create placeholder, placeholder scripts and templates suitable for your application.

3. **priint:cometserver / Whiteboard, direct link to PIM DB**

    Install everything but *Product Module* and *Product Data*, if no InDesignServer is installed also omit *InDesignServer Configuration*.

    Next step will be to create views on your PIM tables in comet_config, configure panelstatements to access product data and create placeholders, placeholder scripts and templates suitable for your application. Note that predefined setups are available for certain system vendors, if in doubt please contact support@priint.com.

4. **priint:comet Desktop**

    See (2.) or (3.). You can omit *CoreSystemServer*, *Privileges* and *InDesignServer Configuration*.

5. **priint:cometserver as InDesignServer LB / Session management server**

    If you use priint:cometserver mainly to address InDesignServer, you only have to install *CoreSystemServer* and *InDesignServer Configuration*.

See Appendix "**Fehler! Verweisquelle konnte nicht gefunden werden. Fehler! Verweisquelle konnte nicht gefunden werden.**" for a detailed list of databases, tables and data imported with each component.

3.1.6.1 CoreSystem

This will install the comet_config database schema and core panelstatements. These components are required for most priint:comet applications (unless you plan a "barebone" priint:cometserver installation). After installing the CoreSystem you will be able to use the priint:workbench for further configuration or install additional components.

3.1.6.2 CoreSystemServer

This will install the comet_admin database schema and core data. The admin database is required to run priint:cometserver.

After installing you will be able to setup a priint:cometserver for various tasks.

3.1.6.3 Plug-In Support

This will install additional data required to connect to your data source either with InDesign Desktop or InDesign Server Plug-Ins. You should run this installation step to have a standard domain and standard user accounts, otherwise you will have to configure domains and accounts using the workbench or another database editor.

After installing you will be able to use priint:comet Desktop or Server Plug-Ins.

3.1.6.4 Product Module

This will install the comet_data database schema and views for products as well as panelstatements and scripts required to access product data. Note: this will install the comet export database schema and appropriate statements. There are other versions available for certain system vendors, if in doubt please contact support@priint.net.

After installing you will be able to access product data from InDesign Server, InDesign Desktop and priint:cometserver / Whiteboard.

3.1.6.5 Product Data

This will install a set of placeholders, templates and showcase data. Installing Product Data may be a good starting point for custom configurations.

If you want to use the showcase, please perform the additional installation steps in Appendix "D Showcase Installation".

After installing, you will be able to place products and generate pages with InDesign Desktop, InDesign Server and priint:cometserver / Whiteboard.

3.1.6.6 Publication Module

This will install the comet_data database schema and views for publications as well as panelstatements and scripts required to manage publications.

After installing you will be able to use the Publication Panel in InDesign Desktop and priint:cometserver / Publication Planner.

3.1.6.7 Publication Data

This will install some configuration data required to work with the Publication Planner (such as publication types). It is recommended to install this data as a starting point for further configuration.

After installing you will be able to work with priint:cometserver / Publication Planner.

3.1.6.8 Workflow Management Module

This will install the comet_data database schema and views for Workflow Management.

After installing you will be able to use the priint:suite Workflow Management Module.

3.1.6.9 Workflow Management Data

This will install some configuration data required to work with the Workflow Management Module. It is recommended to install this data as a starting point for further configuration.

After installing you will be able to work with priint:suite Workflow Management Module.

3.1.6.10 Privileges

This will install priint:suite and Whiteboard privileges. Though it is possible to setup the privileges from scratch, it is highly recommended to run this installation step for all priint:cometserver purposes as a starting point for further adaptions.

After installing you will be able to login to the priint:suite, use the Publication Planner and other priint:cometserver applications.

3.1.6.11 InDesignServer Configuration

This will install a standard configuration to access InDesign Server from priint:cometserver. It is recommended to run this installation step, the configuration can easily be adapted to your needs later.

After installing you will be able to use InDesign Server / priint:comet functions with priint:cometserver.

3.1.7   Standard Accounts

Login	Password	Database	Purpose
Admin	priintadmin	comet_admin	**priint:suite** Administrative tasks
Demo	priintdemo	comet_config	**priint:suite** Typical PublicationPlanner / Whiteboard user **InDesign Desktop**
Import	priintimport	-	**priint:suite** Data import tasks
Idserver	priintidserver	comet_config	**InDesignServer** Global account for InDesignServer data connections
Docuuser	priintdocuuser	comet_config	**priint:suite** All privileges and rights in the priint:suite PublicationPlanner and Whiteboard **InDesign Desktop**
docuuserwf	priintdocuuserwf	comet_config	**priint:suite** All privileges and rights in the priint:suite Workflow module
Pm	priintpm	-	**priint:suite** Typical setup for product manager.

3.2   SOAP WebService

Resources for WebServices requirements for the priint:comet system can be found in the Plug-In documentation and priint:cometserver documentation.

More information can be requested from support@priint.net.

3.3   XML Offline

Please request for a XML Offline tutorial project, if you intend to use XML files as the datasource for a priint:comet system. Basically XML Offline is intended for small projects, evaluation and proof of concepts. Also sometimes an offline project can be useful to deliver test cases.

4   Adobe InDesign / priint:comet Plug-Ins

4.1   Requirements

priint:comet Desktop Plug-Ins are available for CS4, CS5 and CS5.5. Please respect the system requirements as stated by Adobe. In addition, depending on your application, you must also consider the scaling requirements available from Werk II.

4.1.1   General Hints

The priint:comet Desktop Plug-Ins save data in several folders

- log files: usually the users *Documents* folder, e.g. /Users/me/Documents/
- cache: usually a folder named *XCache* within *Documents*
- Login history and publication status: the Plug-In installation directory, e.g.
  C:\Program Files\Adobe\Adobe InDesign CS5\Plug-Ins\Werk II R3508

Log file and cache path can be changed in InDesign using the "Plug-Ins > Write log file…" respective "Plug-Ins > Internal file cache …" command.

Please make sure, that

- neither of these folders is included for real time virus scanning or, vice versa
- either of these folders is excluded from real time virus scanning
- all these folders (including the Plug-Ins folder) are writeable for the user working with InDesign
- none of these folders is a network mount

! *XCache* folders and log files **must never** be used by more than one InDesign Instance (either Desktop or Server) at a time. It is highly recommended using local paths.

If in doubt, ask your system administrator.

4.1.2   Windows

No additional preparation required.

4.1.3   Apple Mac OS X

If you want to connect to an ODBC data source, you have to install the iODBC Framework, which can be downloaded from

<div align="center">

http://www.iodbc.org/dataspace/iodbc/wiki/iODBC/Downloads

</div>

Scroll down to the Mac OS X section and select the installation package suitable for your OS version. For Mac OS X 10.6, 10.7 and 10.8 you can use the 10.5 package.

Alternatively you can install any of the ODBC drivers available from Openlink, this includes installation of the framework. For more information visit

<div align="center">

http://www.openlinksw.com

</div>

You must install the iODBC Framework if you install either the *CoreService* or *CoreService [Database]* Plug-In, otherwise these Plug-Ins cannot be started. *CoreService [Internet]*, *CoreService [XML]*, *CoreService [Oracle]* and *CoreServlett* can be started without the iODBC Framework.

4.2 Installing priint:comet Plug-Ins

4.2.1 Select Appropriate Plug-Ins

Usually the Plug-Ins installer will do this job for you, but in case you have to install a hotfix (which are sometimes delivered as a ZIP archive) or you cannot use the installer for any reason, please refer to Appendix "A.1 priint:comet Desktop Plug-Ins".

4.2.2 Installation With Plug-In Installer

We recommend using the installer provided by Werk II, such as

- **Windows**: priint.comet CS5 V3.3 RC R3508 setup.exe
- **Mac**: priint.comet_CS5_3.3_R3508 Installer.app.zip

If required unpack the installer, double click the installer icon and follow the instruction on the screen.

4.2.3 Manual Installation

To manually install Plug-Ins you need a Plug-In ZIP archive, such as

- **Windows**: Comet 3.3 R3508 (Windows), CS5.zip
- **Mac**: Comet 3.3 R3508 (Mac), CS5.zip

Installation procedure is as follows:

1. shutdown InDesign
2. unpack the archive
3. **new installation**
   - move the resulting folder to the Adobe Plug-Ins directory (e.g. C:\Program Files(x86)\Adobe\Adobe InDesign CS5\Plug-Ins)
   - remove unneeded Plug-Ins. Please refer to Appendix "A.1 priint:comet Desktop Plug-Ins" to decide, which of the Plug-Ins are required.
   - on **Windows** you have to remove both the Plug-Ins and corresponding resource directories
4. **update an existing installation**
   - manually replace the Plug-Ins in the priint:comet Plug-Ins folder (e.g. *C:\Program Files(x86)\Adobe\Adobe InDesign CS5\Plug-Ins\Werk II (3.3)*) with the Plug-Ins just unpacked. Only copy the Plug-Ins that have been installed before
   - on Windows you have to copy both the Plug-Ins and corresponding resource directories
   - take care to preserve license file (w2.lic), login history (xloginhistory.xml), publication status (if available) and other files in the priint:comet Plug-Ins folder, which are not part of the installation package.

Mac: Please use the *Archive Utility* application to unpack the Plug-Ins or the *ditto* command if you unpack in a command line (do not use unzip!), e.g.

```
ditto -xk 'Comet 3.3 R3508 (Mac), CS5.zip' .
```

In this case *ditto* creates a folder named *Werk II R3508* and unpacks all Plug-Ins in this folder.

4.3 Licensing priint:comet Desktop Plug-Ins

You can use the Plug-Ins without license for a test period of 30 days. To order a license, select "About Plug-Ins > Order Service …" (on Mac in the "InDesign" menu, on Windows in the "Help" menu), generate a license request and send it to license@priint.com.

In return you will receive a license file. Just save this file in the priint:comet Plug-Ins folder (next to the priint:comet Plug-Ins) and restart InDesign. Please contact license@priint.com for pricing information also.

5 Adobe InDesignServer / priint:comet Server Plug-Ins

5.1 Requirements

priint:comet Server Plug-Ins are available for CS4, CS5 and CS5.5, for Mac and for Windows / InDesignServer 32bit or Windows / InDesignServer 64bit. Please respect the system requirements as stated by Adobe. In addition, depending on your application, you must also consider the scaling requirements available from Werk II.

5.1.1 Installation Hints For Adobe InDesignServer

Install InDesignServer according to the Adobe InDesignServer documentation.
Depending on the version you will have to enter a serial number upon installation or when first starting the server. If the latter is the case, start InDesignServer *ONCE* from a command line with the "–serialnumber" option, e.g. (paths will probably differ for your installation / CS version):

Windows: in the command prompt type (finish each line with [Enter])

```
cd C:\Program Files(x86)\Adobe\Adobe InDesign CS5 Server\
InDesignServer.com -serialnumber XXXX-XXXX-XXXX-XXXX-XXXX-XXXX
```

Mac: in a Terminal window type (finish each line with [Enter])

```
cd /Applications/Adobe\ InDesign\ CS5\ Server/
./InDesignServer -serialnumber XXXX-XXXX-XXXX-XXXX-XXXX-XXXX
```

5.1.2 General Hints

The priint:comet Server Plug-Ins save data in several folders

- log files: the common comet log, batch log and API log
- cache
- CORBA ior file

The destination paths can be set with several command line arguments, see Appendix "B InDesignServer Command Line Arguments". Please make sure, that

- the destination folders exist and are writeable for the InDesignServer user
- neither of these folders is included for real time virus scanning or, vice versa
- either of these folders is excluded from real time virus scanning
- none of these folders is a network mount. If in doubt, ask your system administrator.

XCache folders and log files **must never** be used by more than one InDesign Instance (either Desktop or Server) at a time. It is highly recommended using local paths.

5.1.3 Fonts, PDF Profiles ("Joboptions"), Dictionaries And InDesignServer

Beside the OS font system, fonts **must** be installed in the *Fonts* directory in the InDesignServer installation directory. Please refer to the Adobe InDesignServer documentation for more information.

If you cannot use PDF Profiles ("Joboptions") or dictionaries with InDesignServer, this might just be a system provileges / account issue. Depending on the OS, these settings are saved in global / default and / or user directories, such as

- C:\ProgramData\Adobe\Adobe PDF
- C:\Users\Administrator\AppData\Roaming\Adobe\Adobe PDF\Settings
- /Library/Application Support/Adobe/Adobe PDF
- /Library/Application Support/Adobe/Adobe Linguistics

Usually you just have to copy these folders into your "local" (i.e. user) settings folder. Please see

http://helpx.adobe.com/content/dam/kb/en/407/kb407987/attachments/ID_Presets___Prefs.pdf

for more information.

5.1.4 Windows

No additional preparation required.

5.1.5 Apple Mac OS X

If you want to connect to an ODBC data source, you have to install the iODBC framework, which can be downloaded from

http://www.iodbc.org/dataspace/iodbc/wiki/iODBC/Downloads

Scroll down to the Mac OS X section and select the installation package suitable for your OS version. For Mac OS X 10.6, 10.7 and 10.8 you can use the 10.5 package.

Alternatively you can install any of the ODBC drivers available from Openlink, this includes installation of the framework. For more information visit

http://www.openlinksw.com

You must install the iODBC Framework if you install the *CoreService[Server]*, otherwise this Plug-Ins cannot be started. *CoreService [Internet][Server]*, *CoreService [Oracle][Server]* and *CoreServlett[Server] can* be started without the iODBC Framework.

5.2 Installing priint:comet Server Plug-Ins

5.2.1 Select Appropriate Plug-Ins

priint:comet Server Plug-Ins are delivered as ZIP archives. In addition to the priint:comet Plug-Ins (which you might already be familiar with from Desktop installations), you should have received a set of Connector Plug-Ins, e.g. Comet 3.3 Connector R3508 (Mac), CS5(Server).zip.

Please do refer to Appendix "A.2 priint:comet Server Plug-Ins" to decide, which Plug-Ins are is suitable for your application and required to connect InDesignServer with priint:cometserver.

5.2.2 Installation

To install Plug-Ins you need a Plug-In ZIP archive, such as

- **Windows**: Comet 3.3 R3508 (Windows), CS5(Server).zip or Comet 3.3 R3508 (Windows), CS5(Server-x64).zip
- **Mac**: Comet 3.3 R3508 (Mac), CS5(Server).zip

Installation procedure is as follows:

1. shutdown InDesignServer
2. unpack the Plug-Ins archive

3. **new installation**
  - move the resulting folder to the Adobe Plug-Ins directory (e.g. *C:\Program Files(x86)\Adobe\Adobe InDesign CS5 Server\Plug-Ins*)
  - remove unneeded Plug-Ins. Please refer to Appendix "A.2 priint:comet Server Plug-Ins" to decide, which of the Plug-Ins are required.
  - on **Windows** you have to remove both the Plug-Ins and corresponding resource directories

4. **update an existing installation**
  - manually replace the Plug-Ins in the priint:comet Plug-Ins folder (e.g. *C:\Program Files(x86)\Adobe\Adobe InDesign CS5 Server\Plug-Ins\Werk II (3.3)*) with the Plug-Ins just unpacked. Only copy the Plug-Ins that have been installed before
  - on **Windows** you have to copy both the Plug-Ins and corresponding resource directories
  - take care to preserve license file (w2.lic) and other files in the priint:comet Plug-Ins folder, which are not part of the installation package.

! **Mac:** Please use the *Archive Utility* application to unpack the Plug-Ins or the *ditto* command if you unpack in a command line (do not use unzip!), e.g.

```
ditto -xk 'Comet 3.3 R3508 (Mac)(Server), CS5.zip' .
```

In this case the *ditto* creates a folder named *Werk II R3508* and unpacks all Plug-Ins in this folder.

5.2.3 Installing Connector Plug-Ins

To install the Connector Plug-Ins you need a Plug-In ZIP archive, such as

- **Windows**: Comet 3.3 Connector R3508 (Windows), CS5(Server).zip or Comet 3.3 Connector R3508 (Windows), CS5(Server-x64).zip
- **Mac**: Comet 3.3 Connector R3508 (Mac), CS5(Server).zip

Installation procedure is as follows:

1. shutdown InDesignServer
2. unpack the Connector Plug-Ins archive
3. **new installation**
  - move the resulting folder to the Adobe Plug-Ins directory (e.g. *C:\Program Files(x86)\Adobe\Adobe InDesign CS5 Server\Plug-Ins*)
  - remove the unneeded Plug-In. Please refer to Appendix "A.2 priint:comet Server Plug-Ins" to decide, which of the Connector Plug-Ins you need.
  - on **Windows** you have to remove both the Plug-Ins and corresponding resource directories
4. **update an existing installation**
  - manually replace the Plug-Ins in the priint:comet Plug-Ins folder (e.g. *C:\Program Files(x86)\Adobe\Adobe InDesign CS5 Server\Plug-Ins\Werk II Connector (3.3)*) with the Plug-Ins just unpacked. Only copy the Plug-Ins that have been installed before
  - on **Windows** you have to copy both the Plug-Ins and corresponding resource directories

Note: starting with CS6 Adobe has dropped Corba Support for InDesignServer. Therefore we wont deliver the Corba Support Plug-In for CS6 and decided to drop support for all CS versions starting with **Plug-In Version 3.4**. You can use the *Soap Support* Plug-In instead, this will be supported in all CS and Plug-In versions. You can safely install this Plug-In, if you don't activate it using proper command line arguments, it will do just nothing.

Mac: Please use the *Archive Utility* application to unpack the Plug-Ins or the *ditto* command if you unpack in a command line (do not use unzip!), e.g.

```
ditto -xk 'Comet 3.3 R3508 (Mac)(Server), CS5.zip' .
```

In this case *ditto* creates a folder named *Werk II R3508* and unpacks all Plug-Ins in this folder.

5.3 Licensing priint:comet Server Plug-Ins

There is no test period for the priint:comet Server Plug-Ins, so you must have a valid license to use the priint:comet Server Plug-Ins capabilities.

1. After installing the Plug-Ins start InDesignServer from a command line.
   - **Windows:** in the command prompt type (finish each line with [Enter])
   ```
   cd C:\Program Files(x86)\Adobe\Adobe InDesign CS5 Server\
   InDesignServer.com
   ```
   - **Mac:** in a Terminal window type (finish each line with [Enter])
   ```
   cd /Applications/Adobe\ InDesign\ CS5\ Server/
   ./InDesignServer
   ```

   (depending on the installation location and CS version the path may differ).

2. After a while you should see a message like

```
. NO LICENSE FILE FOUND.
.
. The plugins will only work as reader. If the full functionality
. of priint.comet is needed, you need a valid license from Werk II.
. If you have a valid license file w2.lic,
. please place this file into the Plug-Ins folder or restart
. InDesign Server with the -cometlic <path to w2.lic> option.
.
. If you do not have a valid license, then for ordering please
. copy and paste the following lines and send them to
.
. license@werk-ii.com
.
.
"InDesign Server CS5"
"11550003567098748020"
"00:26:BB:61:4A:F2"
"3.3"
CometBatch
Adaption
```

3. As instructed on the screen, copy and paste the last five lines and send them to license@werk-ii.com (for the sake of convenience you can as well use license@priint.com).
   In return you will receive a license file. Just save this file in the priint:comet Plug-Ins folder (next to the priint:comet Server Plug-Ins) and restart InDesignServer. Please contact

license@priint.com for pricing information also.

See Section "5.4 InDesignServer / Plug-Ins Command Line Arguments", if you want to keep the license file in a different location.

5.4 InDesignServer / Plug-Ins Command Line Arguments

See Appendix "B InDesignServer Command Line Arguments" for a list of available command line arguments.

These are some recommendations for typical situations (type all commands in one line):

InDesignServer + priint:cometserver, connecting with CORBA (Windows)

```
InDesignServer.com -configuration Instance1 -iorfile C:\comet\ids1.ior
-cometcache C:\comet\cache\instance1 -cometlog C:\comet\log\comet1.log
-cometapilog C:\comet\log\api1.log -cometlic C:\comet\w2.lic -previews
-LogToApplicationEventLog
```

InDesignServer + priint:cometserver, connecting with CORBA (Mac)

```
./InDesignServer -configuration Instance1 -iorfile /var/comet/ids1.ior
-cometcache /var/comet/cache/instance1 –cometlog
/var/comet/log/comet.log -cometapilog /var/comet/log/api.log -cometlic
/var/comet/w2.lic -previews
```

InDesignServer + priint:cometserver, 2nd instance connecting with CORBA (Windows)

```
InDesignServer.com -configuration Instance2 -iorfile C:\comet\ids2.ior
-cometcache C:\comet\cache\instance2 -cometlog C:\comet\log\comet2.log
-cometapilog C:\comet\log\api2.log -cometlic C:\comet\w2.lic -previews
-LogToApplicationEventLog
```

InDesignServer + priint:cometserver, connecting with SOAP (Mac)

```
./InDesignServer -configuration Instance1 –cometport 4711 -cometcache
/var/comet/cache/instance1 -cometlog /var/comet/log/comet1.log -
cometapilog /var/comet/log/api1.log -cometlic /var/comet/w2.lic -
previews 2>&1>> /var/comet/log/ids1.log
```

InDesignServer in Batch mode (Autocomet) (Windows)

```
InDesignServer.com -configuration Instance1 -cometconfig
C:\comet\config.xml -cometcache C:\comet\cache\instance1 -cometlog
C:\comet\log\comet1.log -cometapilog C:\comet\log\api1.log -cometlic
C:\comet\w2.lic -previews -LogToApplicationEventLog
```

You should *enable* logging (i.e. provide the cometlog and cometapilog argument) while developing and configuring your project.
You should *disable* logging (i.e. provide the cometlog and cometapilog argument) in production environments to achieve better performance. Logging is disabled by just omitting the *cometlog* and *cometapilog* command line arguments.

5.5 Running InDesignServer in Batch Mode (Autocomet)

Batch / Autocomet mode allows asynchronous job processing. Multiple InDesignServer and InDesignServer instances can be connected to one job source.

Though nowadays you would rather use priint:cometserver to control and load balance InDesignServer, here is a short description how to setup InDesignServer to run in batch / Autocomet mode.

To enable batch processing, either

- put a file named config.xml in the priint:comet Plug-Ins folder (e.g. *C:\Program Files(x86)\Adobe\Adobe InDesign CS5 Server\Plug-Ins\Werk II (3.3)*), or
- start InDesignServer with the command line option
  ```
  -cometconfig <path to config.xml>
  ```

The config.xml contains connection data, type of batch processing, polling interval etc. See *Resources/Autocomet.zip* for an example.

Batch mode requires the implementation of the Autocomet interface in your data source. For database connection this is the comet_config.dmn_* tables, for SOAP connections this is the CometBatch interface resp. the CometBatch WSDL. Refer to the Plug-In or server development documentation for further details.

5.6 Connecting InDesignServer With priint:cometserver

To connect InDesignServer with priint:cometserver you must

- configure priint:cometserver to register InDesignServer instances
- start InDesignServer with either *Corba Support* or *Soap Support* enabled

To enable *Corba Support*

1. install the *Corba Support* Plug-In (part of the Connector Plug-In archive)
2. start InDesignServer with the

   ```
   -iorfile <path to iorfile>
   ```

   option.

   To make it obvious, the startup command would be something like (e.g. Windows)

   ```
   InDesignServer.com -iorfile C:\comet\ids1.ior
   ```

If you use **CORBA** make sure, the iorfile is writeable for the InDesignServer user. The *iorfile* option must match the priint:cometserver endpoint configuration for this particular InDesignServer instance.

CORBA supports local connections only (this is a limitation of the Adobe CORBA implementation), therefore InDesignServer and priint:cometserver must be installed on the same machine, if you want to connect using CORBA.

To enable *Soap Support*

1. install the *Soap Support* Plug-In (part of the Connector Plug-In archive)
2. start InDesignServer with the

   ```
   -cometport <listen port>
   ```

   option.

   To make it obvious, the startup command would be something like (e.g. Mac)

   ```
   ./InDesignServer -cometport 4711
   ```

If you use **SOAP** make sure, the port isn't used by another application, blocked by a firewall or for any other reason not available for the user. A common mistake is trying to bind privileged ports (those <= 1024) with normal user privileges, which is not allowed on most Unix systems. The *cometport* option must match the priint:cometserver endpoint configuration for this particular InDesignServer instance.

If you imported the *InDesignServer Configuration* database module (see Section "3.1.6.11 InDesignServer Configuration"), your system is already configured to register *one* InDesignServer instance connected via CORBA.

See Section "6.3.5 Configure InDesignServer Instances" for details how to manage InDesignServer instances with priint:cometserver.

5.7 Start / Stop InDesignServer

5.7.1 Considerations

You can start and stop InDesignServer manually or as a service. There is no general advantage of one way over the other (once running, InDesignServer and the priint:comet Server Plug-Ins support all features, whether you started InDesignServer as a service or manually), so, which way is more suitable for you primarily depends on the intended use.

Service

- better availability
- easier system startup. Once configured, you don't have to bother about InDesignServer startup anymore
- more robust, harder to compromise

Manually Start / Stop

- easier to setup
- easier to shutdown / disable
- easier to change configuration (startup parameters, Plug-Ins)
- easier to debug. If InDesignServer crashes, you will notice sooner or later. You might miss that, if InDesignServer is silently restarted in the background.

As a general rule, I would suggest to

- use services for integration tests and production environments
- use services if installed on a VM dedicated for demo purposes
- manually start / stop InDesignServer during development and testing phase
- manually start / stop InDesignServer on mobile devices (such as marketing laptops)

5.7.2 Manually Start / Stop InDesignServer On Microsoft Windows

Open a command prompt, change to the InDesignServer installation directory

```
cd C:\Program Files\Adobe\Adobe InDesign CS5 Server\
```

Start a InDesignServer instance (type all in one line)

```
InDesignServer.com -configuration instance1 -cometcache
C:\comet\cache\instance1
```

To stop this instance hit *CTRL-c* in the console window (InDesignServer will process the signal when idle the next time, so shutdown may take a while). If it's not possible to shutdown InDesignServer in the console window, kill the process in the Windows Taskmanager.

To start a second or third InDesignServer instance open another command prompt, cd to the installation directory and type (for example, type all in one line)

```
InDesignServer.com -configuration instance2 -cometcache
C:\comet\cache\instance2
```

Remember, that configuration and cometcache must be unique for each instance running on one server.

5.7.3 Manually Start / Stop InDesignServer On Apple Mac OS X

Open a Terminal window, change to the InDesignServer installation directory

```
cd /Applications/Adobe\ InDesign\ CS5\ Server/
```

Start a InDesignServer instance

```
./InDesignServer -configuration instance1 -cometcache
/var/comet/cache/instance1
```

To stop this instance hit *CTRL-c* in the Terminal window (InDesignServer will process the signal when idle the next time, so shutdown may take a while). If it's not possible to shutdown InDesignServer in the Terminal window, find out the process id with

```
ps a|grep -i [i]ndesignserver
```

This will output something like

```
  2805 s000  S+     0:00.00 /bin/bash ./InDesignServer -iorfile
/var/comet/ids1.ior -cometcache /var/comet/cache/instance1
  2807 s000  S+     0:05.27 ./InDesignServer.app/Contents/MacOS/InDesignServer
 -iorfile /var/comet/ids1.ior -cometcache /var/comet/cache/instance1
```

The first number is the process id of the InDesignServer process, in this case you would terminate InDesignServer with

```
kill -9 2805
```

To start a second or third InDesignServer instance open another Terminal window, cd to the installation directory and type (for example)

```
./InDesignServer -configuration instance2 -cometcache
/var/comet/cache/instance2
```

Remember, that configuration and cometcache must be unique for each instance running on one server.

5.7.4 Running InDesignServer As A Windows Service

If during the installation the component "ID SERVER SERVICE FILES" (one each for 32bit and 64bit) was selected, the installer will register two Windows services that can start and monitor InDesignServer instances without an interactive user session login.

Parallel installation of the 32bit and 64bit version should be avoided. If you have installed both services, make sure that the one you are not going to use is set to disabled in the Windows services configuration dialogue.

You can configure, add and remove instances by using the Windows Management console plugin. It can be found in the InDesign Server installation directory as *InDesignServerService.msc*. To launch the Windows Management console, just double click the *.msc* file.

This is how it should work, below you find an explanation how to fix problems, if it doesn't work right out if the box (which in our experience never does).

1. double click the *InDesignServerService.msc* file in the InDesignServer installation directory. This will launch the Windows Management console.

2. right click on the *InDesignServerService* entry in the left tree view. Select *New > New InDesign*

Server instance. A new entry will show up in the list view. A double click will open the configuration dialogue for this instance. The configuration is equivalent to the command line configuration mentioned above

- leave the port number empty (neither batch processing nor priint:cometserver connection require Adobes built in SOAP service)
- type in all command line arguments in the second input field. Refer to Section "5.4 InDesignServer / Plug-Ins Command Line Arguments" for details.

3. open the service settings in the *Windows service administration* tool and under the *Log On* assign an account that's equipped with the proper privileges to run InDesignServer.

4. restart the InDesignServerService service to activate new instances or new settings

By using the process view in the Windows Taskmanager, you should see one InDesignServer process for each instance alongside one single InDesignServerService (or two if you run 32 and 64 bit services parallel).

For more details please refer to Adobe's "Introduction to Adobe InDesign CS5 Server" documentation, e.g. chapter "Configuring InDesign Server Windows Service".

Windows Management console error

In some cases the management console plugin needs manual fixes to work properly. If double clicking the *.msc* file results in an error message open a command prompt with administrator privileges, navigate to your InDesign Server installation directory (e.g. *C:\Program Files (x86)\Adobe\Adobe InDesign CS5 Server*) and enter the following command

```
regsvr32 InDesignServerMMC64.dll /u
```

to remove the damaged registration and reinstall it with

```
regsvr32 InDesignServerMMC64.dll
```

The "**64**" in the file name refers to the operating system, not the InDesignServer. On a **32bit** OS reinstalling the registration goes like

```
regsvr32 InDesignServerMMC.dll /u
regsvr32 InDesignServerMMC.dll
```

After this, you should be able to launch the management console plugin.

Error #1053

If InDesign Server Service does not start properly and exits with an error *"#1053 The service did not respond to the start or control request in a timely fashion."*, there are two ways to fix the problem:

1. Does not affect the rest of the system, but doesn't work on all machines

1. open the Windows service configuration dialogue
2. find "InDesignServerService"
3. select "Properties" from the context menu (right click) and under "Recovery" switch "First failure" and "Second failure" to "Restart the Service"

2. Reliable, but affects all services

1. click the **Start** button, then click **Run**, type **regedit**, and click **OK**.

2. in the **Registry Editor**, click the registry subkey

   `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control`

3. in the details pane, locate the **ServicesPipeTimeout** entry, right-click that entry and then select **Modify**. **Note:** If the **ServicesPipeTimeout** entry does not exist, you must create it by selecting **New** on the **Edit** menu, followed by the **DWORD Value**, then typing **ServicesPipeTimeout**, and clicking **Enter**.

4. click **Decimal**, enter the new timeout value in milliseconds, and then click **OK**.

5. restart the computer

The default value is 30000ms (=30s), which seems to be too short for some servers running as a virtual machine or on low hardware resources.

This instruction was adapted from the Microsoft Technet WebSite, more information can be found at http://technet.microsoft.com/en-us/library/dd349403%28v=ws.10%29.aspx.

There are also cases where de- and manually re-registering the service is necessary if the above steps do not help. To do that

1. open the command prompt with administrator privileges

2. navigate to your InDesign Server installation directory (e.g. *C:\Program Files (x86)\Adobe\Adobe InDesign CS5 Server*)find "InDesignServerService"

3. enter the follow command to unregister the service:

   `InDesignServerService /install /u`

4. Reboot the system, with administrator privileges run the following command in the same directory

   `InDesignServerService /install`

5.7.5 Running InDesignServer As A Mac OS X Service using launchd

To start and run InDesignServer using launchd, you must define a daemon for each InDesignServer instance. To do so, put a configuration file into /Library/LaunchDaemons/. This file should have a reasonable name (so it can be identified for later editing) and must have the extension *.plist*.

Example

```
File: /Library/LaunchDaemons/com.priint.comet1.plist
```

```xml
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE plist PUBLIC ="-//Apple Computer//DTD PLIST 1.0//EN"
"http://www.apple.com/DTDs/PropertyList-1.0.dtd">
<plist version="1.0">
  <dict>
    <key>Label</key>
    <string>com.priint.comet1</string>
    <key>UserName</key>
    <string>priint</string>
    <key>GroupName</key>
    <string>wheel</string>
    <key>OnDemand</key>
    <false/>
    <key>ProgramArguments</key>
    <array>
      <string>/Applications/Adobe InDesign CS5
Server/InDesignServer.app/Contents/MacOS/InDesignServer</string>
      <string>-iorfile</string>
      <string>/var/comet/ids1.ior</string>
      <string>-cometcache</string>
      <string>/var/comet/cache1</string>
      <string>-configuration</string>
      <string>instance1</string>
      <string>-previews</string>
    </array>
    <key>RunAtLoad</key>
    <false/>
  </dict>
</plist>
```

Pretty straightforward. To add more command line arguments, you would simply add more string-element pairs (or single string-elements) to the ProgramArguments array. To add another instance, you just have to copy this file to – say - */Library/LaunchDaemons/com.priint.comet2.plist* and change the *iorfile*, *configuration* and *cometcache* values.

On system startup all daemons found in /Library/LaunchDaemons are started (unless OnDemand is set to *true*, which at the time of this writing doesn't make sense in a priint:comet environment).

To start the service manually, open a Terminal window and type

```
sudo launchctl load /Library/LaunchDaemons/com.priint.comet1.plist
```
To stop manually, use

```
sudo launchctl unload /Library/LaunchDaemons/com.priint.comet1.plist
```

It is not recommended saving Daemon files in the */System/Library/LaunchDaemons* folder, these entries may be deleted on system updates. Daemons in */Library/LaunchDaemons* will be started literally the same way like daemons in */System/Library/LaunchDaemons*.

5.8 Advanced InDesignServer Configuration

You should

- have a look at the *Command Line Arguments* section in Appendix B
- especially consider using the *-shutdowntrigger* option to control automatic maintenance restarts
- have a closer look at the priint:cometserver *preferences* (they can be administered in the comet_admin.serverproperty table or using the priint:suite admin UI).

The **shutdowntrigger** and **purgetrigger** command line options might be worth a closer look:

Shutdown Trigger

Shutdown the InDesignServer instance...

- -shutdowntrigger uptime=nd | nh | nm | ns: after n days / hours / minutes / seconds uptime
- -shutdowntrigger documents=n: after n documents have been processed
- -shutdowntrigger jobs=n: after n jobs have been processed
- -shutdowntrigger memory[=low]: when InDesign reports low memory
- -shutdowntrigger time=*hh:mm:ss*: at a certain time
- -shutdowntrigger headroom=nG | nM | nK: when no more n GB, MB or kB memory can be allocated

You can set this option several times, **Example**:

```
-shutdowntrigger documents=100  -shutdowntrigger 1d
```

will shut down the instance every 100th document or after one day uptime, whichever event occurs first.

If it is not possible to shutdown the instance (e.g. if documents are currently opened) shutdown will be postponed to the next possible moment.

Purge Trigger

Purge memory...

- -purgetrigger uptime=nd | nh | nm | ns: after n days / hours / minutes / seconds uptime
- -purgetrigger documents=n: after n documents have been processed
- -purgetrigger jobs=n: after n jobs have been processed
- -purgetrigger memory[=low]: when InDesign reports low memory
- -purgetrigger time=*hh:mm:ss*: at a certain time
- -purgetrigger headroom=nG | nM | nK: when no more n GB, MB or kB memory can be allocated

This option can be set several times also.

The *purgetrigger* option can be useful for **debugging**, especially if you notice occasional InDesignServer crashes. These are often caused by leaked document resources, which are hard to track down, because InDesignServer will just crash *anytime* when memory is purged (which usually can be hours later).

Setting

```
-purgetrigger uptime=1s
```

is a good test to check your configuration. Though, it's not recommended using purgetrigger in a productive environment.

6   priint:cometserver And Whiteboard

6.1   Requirements

priint:cometserver requires

- a Java SDK 6 (aka Java SE 6 1.6.0), at least update 20
- Oracle GlassFish Server version 2.1.1
- several 3d party libraries, which are delivered with the priint:cometserver installation package

It must be Java 1.6.0, not the most recent version. E.g. Glassfish 2.1.1 is known not to work with Java 1.7. To avoid problems you should deactivate any Java Update agents and make sure, that only the 1.6 SDK is installed.

6.1.1   Java Installation On Windows

Check Java Installation

1. Check, if a Java panel exists in the Windows control panel.
    - if yes, launch the Java panel, select the "Java" tab and click on "View…". This will show the installed Java Runtimes, make sure that at least Version 1.6.0_b20 is installed and enabled
    - if not you must install a Java SDK
2. Open a command prompt, type

    ```
    java –version
    ```

    The output should be something like

    ```
    java version "1.6.0_31"
    Java(TM) SE Runtime Environment (build 1.6.0_31-b04-415-10M3635)
    Java HotSpot(TM) Client VM (build 20.6-b01-415, mixed mode)
    ```

3. In the Windows control panel select "System", change to the "Advanced" tab and click "Environment". Review the *Path* variable; it should contain the path to a Java SDK 1.6.0 (>= update 20, as you ought to know by now). If not, add the path to your Java SDK (e.g. C:\Programme\Java\jdk1.6.0_31\bin\).
4. To triple check the Java installation, you can finally type

    ```
    javac -version
    ```

    in a command prompt, this should output something like

    ```
    javac 1.6.0_31
    ```

If any of the above tests fails, you must install a Java SDK or update your Java installation.

Java SDK Installation

1. go to the Oracle download page
    http://www.oracle.com/technetwork/java/javasebusiness/downloads/java-archive-downloads-javase6-419409.html
2. navigate to Java SE Development Kit 6u31
3. select the installation package suitable for your platform and architecture and run the installer.

6.1.2 Java Installation on Apple Mac OS X

Check Java Installation

1. open a Terminal window and type

   ```
   java -version
   ```

   The output should be something like

   ```
   java version "1.6.0_31"
   Java(TM) SE Runtime Environment (build 1.6.0_31-b04-415-10M3635)
   Java HotSpot(TM) Client VM (build 20.6-b01-415, mixed mode)
   ```

2. launch the Java Preferences Panel, this can be found in */Application/Utilities/*. In the "General" tab you should see a list of installed Java runtime environments.

   Make sure, "Java E 6" is available and activated. To avoid problems, the "Java SE 6" entries (there may be several for 32 and 64bit) must be the default (i.e.: appear at the top of the list).

Java SDK Installation

1. Download the latest Java SE 6 release for OS X:
   - **Mac OS X 10.5** (Leopard): http://support.apple.com/kb/DL1359
   - **Mac OS X 10.6** (Snow Leopard): http://support.apple.com/kb/DL1516
   - **Mac OS X 10.7** (Lion): http://support.apple.com/kb/DL1515
   - **Mac OS X 10.**8 (Mountain Lion?): download Java from the Oracle website
2. Double click the downloaded disk image and run the installer

6.2 Oracle GlassFish Server Installation And Configuration

Before starting the installation, it is essential to consider about

- **installation location**: we recommend installing GlassFish at *C:\glassfish* (Windows) resp. */glassfish/* (Mac)
- **user and system privileges**: The following steps should be executed under the user account intended to run glassfish later.

GlassFish server binds to several ports to listen for requests. Binding to ports (particularly privileged ports on Unix systems, i.e. those <= 1024) may require certain privileges; also these ports must not be blocked by a firewall or used by other applications. If GlassFish cannot bind to the configured ports, the server startup will fail.

There are several ways to work around port binding problems, please search the web for e.g. "mac os x binding to privileged ports without root" for more information

6.2.1 GlassFish Installation

1. download the Oracle GlassFish Server 2.1.1 from

   http://glassfish.java.net/downloads/v2.1.1-final.html

   Choose the binary suitable for your OS / platform. We recommend downloading the English version (not the multi language). If you prefer a graphical installer for Windows, you can request a copy from Werk II. The graphical installer isn't available for download anymore and

therefore may not be supported by Oracle.

2. open a Terminal window or command prompt, change to the download directory and type

   ▪ **Windows**

   ```
   java -Xmx1024m -jar glassfish-installer-v2.1.1-b31g-windows.jar
   ```

   ▪ **Mac**

   ```
   java -Xmx1024m -jar glassfish-installer-v2.1.1-b31g-darwin.jar
   ```

   This will unpack the jar archive and create a folder named *glassfish*.

3. move the glassfish folder to the destination path. In the same Terminal or command prompt type

   ▪ **Windows**

   ```
   move glassfish C:\
   ```

   ▪ **Mac**

   ```
   mv glassfish /
   ```

4. run the ant setup script. This will configure the environment according to the installation location and create the default domain. In a Terminal window or command type

   ▪ **Windows**

   ```
   cd C:\glassfish
   lib\ant\bin\ant -f setup.xml
   ```

   ▪ **Mac**

   ```
   cd /glassfish
   chmod -R +x lib/ant/bin
   lib/ant/bin/ant -f setup.xml
   ```

So far we are done with GlassFish installation. Before we can continue with priint:cometserver installation, some configuration must be fixed.

6.2.2 Required GlassFish Configuration

1. start GlassFish. If in doubt how to do so, refer to Section "6.5.2 Manually Start / Stop GlassFish On Microsoft Windows" or "6.5.3 Manually Start / Stop GlassFish On Apple Mac OS X".

2. login to the GlassFish administration console. Use a internet browser and enter the address http://localhost:4848/ (or http://ip.address:4848/ if you login from a different host). The default username / password is *admin* / *adminadmin*.

3. in the tree view on the left side navigate to *Application Server*, select *JVM Settings* in the right panel and after that *JVM Options* in the tab panel below.

4. change the following JVM options

   ▪ -XX:MaxPermSize=512m (default is -XX:MaxPermSize=192m)

   ▪ -Xmx2048m (default is -Xmx512m)

   click *Save* at the top of the page to apply these settings.

5. select *Administrator Password* in the tab panel at the top to change the admin password

6. in the tree view on the left side navigate to *Configuration > EJB Container*, select EJB Timer Service in the right panel

7. change the value for *Minimum Delivery Interval* to 500 ms, click *Save* to apply this setting.

8. Logout from the Server Administration Console and shutdown GlassFish

The settings on this page generally improve performance and stability, optimal settings may slightly differ for your environment. On a first run you can skip this step and refer to it later.

6.2.3   Recommended JVM Configuration

1. login to the GlassFish administration console (http://localhost:4848)
2. in the tree view on the left side navigate to *Application Server*, select *JVM Settings* in the right panel and after that *JVM Options* in the tab panel below.
3. add the following JVM options (use the *Add JVM Option* button at the top of the list)
    - -Xms2048m
    - -Xmn1024m
    - -XX:+AggressiveOpts
    - -XX:ParallelGCThreads=16
    - -XX:+UseParallelOldGC
    - -XX:LargePageSizeInBytes=256m
    - -XX:+UseParallelGC
    - -Djava.awt.headless=true
    - -server

    Click *Save* to apply the new values

6.2.4   Recommended HTTP Configuration

1. login to the GlassFish administration console (http://localhost:4848)
2. in the tree view on the left side navigate to *Configuration > HTTP Service*, select *Request Processing* in the right panel
3. change the following settings. The following suggestions should serve for general purpose:
    - **Thread Count**: 64
    - **Initial Thread Count**: 16
    - **Thread Increment**: 4
    - **Request Timeout**: 300
    - **Buffer Length**: 65536

    Click *Save* to apply the new values
4. select *HTTP File Cache* in the right panel
5. change the following settings
    - **Globally**: Enabled (at the top of the page)
    - **File Caching Enabled**: ON (at the bottom of the page)

    Click *Save* to apply the new values
6. navigate to *Configuration > HTTP Service > HTTP Listeners > http-listeners-1*
7. on this page you can set the address and port to listen for HTTP requests.

    Also it makes sense to set the number of *Acceptor Threads* to 2 (do not use higher values).
8. the same way edit *http-listener-2* (SSL) and *admin-listener* (default 4848 for the admin console)

6.3   CometServerLight Enterprise Application Installation

Installation of the priint:cometserver application requires some further preparation, which is explained in this section.

6.3.1   Required Libraries, Support Files And Directories

Follow these installation instructions:

1. Unpack the file *Server/3dpartylibs.zip* and move all jar files from the folder just created to the domain **lib** directory (move only the jar files, not the folder itself).
   On Windows, if GlassFish is installed at *C:\glassfish*, this would be
   > *C:\glassfish\domains\domain1\lib*
   on Macintosh analogously
   > */glassfish/domains/domain1/lib*.

2. Copy the *Server/cometserver.lic* file to the domain **config** directory.
   On Windows, if GlassFish is installed at *C:\glassfish*, this would be
   > *C:\glassfish\domains\domain1\config*
   on Macintosh analogously
   > */glassfish/domains/domain1/config*.
   The *cometserver.lic* file can be used to enable or disable certain priint:suite modules, though in general you would rather do this using the priint:suite privileges and roles administration. However, the file must be present, otherwise all priint:suite modules are disabled by default.

3. Unpack the file *Server/CometEnvironment.zip* and move the folder just created to the domain **config** directory.

4. in the domain config directory create two folders named *Documents* and *tmp*. This is the default destination for uploaded documents and the path for temporary session files. These folders must be writeable for the GlassFish user.

When you are done, **restart** or **start** GlassFish (we assume you have not yet setup GlassFish as a service, if you are unsure how to start GlassFish manually, refer to Section "6.5.2 Manually Start / Stop GlassFish On Microsoft Windows" or "6.5.3 Manually Start / Stop GlassFish On Apple Mac OS X").

Make sure, that the domain config directory, the document default path and the temporary path are

- not included for real time virus scanning or, vice versa
- excluded from real time virus scanning
- writeable for the GlassFish user

The config and temporary path should not be a network mount either.

Change Documents And Temporary Paths

You can change the document default and temporary path in the comet_admin.serverproperty table. The following SQL statement

```
SELECT * from comet_admin.serverproperty WHERE
propertyKey='documentRootPath' OR propertyKey='tmpPath';
```

should output something like

```
+--------+----------------+--------------+--------------+
| id     | propertyKey    | propertyValue | propertyType |
+--------+----------------+--------------+--------------+
| 270937 | documentRootPath | Documents  | 1            |
| 250151 | tmpPath        | tmp          | 1            |
+--------+----------------+--------------+--------------+
```

You can provide an absolute or relative path here. Relative paths are relative to the domain config directory.

6.3.2 Configure JDBC Resources

priint:cometserver requires at least one JDBC resource named *jdbc/admin* to start up.

- *JDBC resources* are named resources, that can be addressed via JNDI (Java Naming And Directory Interface). JDBC resources are linked with a *connection pool*
- *connection pools* define an actual database connection

So, prior to configuring the JDBC resource you have to configure a connection pool. Since this is dependent of the database vendor, it is documented in Section "7 Data Connectivity".

Setup *jdbc/admin*

1. login to the GlassFish administration console. Use a internet browser and enter the address http://localhost:4848/ (or http://ip.address:4848/ if you login from a different host). The default username / password is *admin* / *adminadmin*.

2. configure a connection pool for the *comet_admin* database, name it *comet_admin*
   Instructions how to configure the connection can be found at
   - **MySQL**: Section "7.1.6 Connect priint:cometserver With MySQL"
   - **SQL Server**: Section "7.2.6 Connect priint:cometserver With SQL Server"
   Oracle: Section "0
   - Connect priint:cometserver With Oracle"

3. navigate to *Resources > JDBC > JDBC Resources*, click *New…* in the right panel. Configure the JDBC resource with the following settings:
   - **JNDI Name**: jdbc/admin
   - **Pool Name**: select the *comet_admin* connection pool
   - **Description**: you may enter a description if you like
   - **Status**: Enabled (checked)

If you intend to use any of the priint:suite modules (Publication Planner, Workflow Management etc.) or the Whiteboard, you need to configure at least on more JDBC resource. This must be named *jdbc/resource1* and is usually associated to *comet_config* database connection.

Again: prior to configuring the JDBC resource you have to configure the corresponding connection pool as documented in Section "7 Data Connectivity".

Setup *jdbc/resource1*

1. login to the GlassFish administration console.

2. configure a connection pool for the *comet_config* database, name it *comet_config*

   Instructions how to configure the connection can be found at

   - **MySQL**: Section "7.1.6 Connect priint:cometserver With MySQL"
   - **SQL Server**: Section "7.2.6 Connect priint:cometserver With SQL Server"

Oracle: Section "0

   - Connect priint:cometserver With Oracle"

3. navigate to *Resources > JDBC > JDBC Resources*, click *New…* in the right panel. Configure the JDBC resource with the following settings:

   - **JNDI Name**: jdbc/resource1
   - **Pool Name**: select the *comet_config* connection pool
   - **Description**: you may enter a description if you like
   - **Status**: Enabled (checked)

The settings above are valid, provided you use a database connection for your application data and your databases are named *comet_admin* resp. *comet_config*.

Why is it jdbc/resource1? Can I have more JDBC resources?

We use the Java Persistence API to access application and configuration data with priint:cometserver. The persistence resources have fixed JNDI names, in our case *jdbc/resource1, jdbc/resource2 … jdbc/resource10*. That means, you can configure up to 10 application data sources (if you need more please contact dev@priint.com).

Upon login you must provide a dataset name. Of course you don't want to enter something like "resource1" in the login dialogue, therefore JNDI resource names are mapped to human understandable names in the *comet_admin.persistenceproperty* table.

```
SELECT persistence_resource,jdbc_resource FROM persistenceproperty;
```

will (in a fresh installation) show something like

```
+----------------------+---------------+
| persistence_resource | jdbc_resource |
+----------------------+---------------+
| resource1            | demo          |
+----------------------+---------------+
```

To change the "public" name of a dataset you only have to

- change the value of the *jdbc_resource* field in the *comet_admin.persistenceproperty* table
- update the corresponding record in the *comet_admin.dataset* table.

To add a dataset (i.e. another human understandable to JNDI name mapping)

- insert a record in *comet_admin.persistenceproperty*, bind it for example to *resource2*
- configure *jdbc/resource2* as described above
- add a record in the *comet_admin.dataset* table

6.3.3 Configure Mail Session

To enable mail delivery you must configure a JavaMail session resource:

1. login to the GlassFish administration console (http://localhost:4848)

2. in the tree view on the left side navigate to *Resources > JavaMail Session*

3. click New… in the right panel and configure a JavaMail Session with the following values
   - **JNDI Name**: mail/gf_mail
   - **Mail Host**: IP address or name of the mail host
   - **Default User**: default user when connecting to the mail server
   - **Default Return Address**: E-Mail address of the default user
   - **Status**: Enabled (checked)

   All other settings can be left unchanged. Finally click *Save* at the top of the page to save the JavaMail Session

6.3.4 Deploy CometServerLight Enterprise Application

After all, if you performed all of the above thoroughly, you can deploy the CometServerLight Enterprise Application (it's *CometServerLight* because the EAR does not contain all the 3dparty libraries you already installed. This will make deployment and update delivering considerable faster. Though: this **is** the full version).

1. Copy the file *Server/CometServerLight.ear* to the domains autodeploy directory. This is on
   - **Windows**: *C:\glassfish\domains\domain1\autodeploy*
   - **Mac**: */glassfish/domains/domain1/autodeploy*

2. After a while you should see a file named *CometServerLight.ear_deployed* in the autodeploy folder.

3. Check the GlassFish server log file for errors. This can be found at
   - **Windows**: *C:\glassfish\domains\domain1\logs\server.log*
   - **Mac**: */glassfish/domains/domain1/logs/server.log*

   At the end of the log file you should see something like

```
[#|2012-05-30T01:02:16.878+0200|INFO|sun-
appserver2.1|javax.enterprise.system.tools.deployment|_ThreadID=17;_ThreadName
=Timer-20;|[AutoDeploy] Successfully autodeployed :
/glassfish/domains/domain1/autodeploy/CometServerLight.ear.|#]
```

6.3.5 Configure InDesignServer Instances

Two tables in the *comet_admin* database keep the InDesignServer configuration. At the time of this writing, there is no UI to manage InDesignServer instances, so you have to add, remove or edit them with a SQL Editor (such as the MySQL Workbench or SQL Server Management Studio).

To get a list of configured InDesignServers, type

```
SELECT * FROM comet_admin.indesignserver;
```
The output should be something like

```
+----------+---------------+---------------+
| serverID | serverName    | serverProtocol |
+----------+---------------+---------------+
| 1        | Local Server  | LOCAL         |
| 2        | Remote Server | REMOTE        |
| 3        | Offline Server | COMETOFFLINE |
+----------+---------------+---------------+
```

To get a list of configured InDesignServer instances, type

```
SELECT * FROM comet_admin.indesignserverinstance;
```
Each InDesignServer instance is defined by

- **instanceID**: a unique ID
- **serverID**: ID of the InDesignServer
- **instanceName**: a name, providing a speaking name will make monitoring easier
- **endpoint**: the communication endpoint. This is in case of
  - CORBA: the ior file path
  - SOAP: service endpoint
  - COMETOFFLINE: the maximum number of parallel threads (!)
- **active**: whether this instance is active (1) or not (0)
- **typeID**: purpose of this InDesignServer instance. This can be one of
  - **1**: process long time running (background) jobs with this instance only
  - **2**: use for any purpose
  - **3**: process Whiteboard requests only
  - **4**: import jobs (not applicable for InDesignServer instances)
  - **5**: use for desktop sessions only
  - **6**: use for API sessions only
  - **7**: use for servlet requests only

  If unsure or only one instance is configured, the typeID should always be set to **2**.

You can configure an arbitrary number of InDesignServers and InDesignServer instances. To do so, launch a SQL Editor or run appropriate SQL statements. After any configuration change you must restart GlassFish.

Attention should be paid to

- not mix COMETOFFLINE with other InDesignServer types
- not configure more instances than are actually available and running

If you imported the *InDesignServer Configuration* database component (see "3.1.6.11 InDesignServer

Configuration"), priint:cometserver is already configured to register one InDesignServer instance connected with CORBA.

If you omitted the *InDesignServer Configuration* component, priint:cometserver is configured to register one "Offline" instance allowing eight parallel threads.

This is the default comet_admin.indesignserver configuration:

```
+----------+----------------+---------------+
| serverID | serverName     | serverProtocol |
+----------+----------------+---------------+
| 1        | Local Server   | LOCAL         |
| 2        | Remote Server  | REMOTE        |
| 3        | Offline Server | COMETOFFLINE  |
+----------+----------------+---------------+
```

A **CORBA** instance configuration looks like this:

```
+------------+----------+--------------+------------------+--------+--------+
| instanceID | serverID | instanceName | endpoint         | active | typeID |
+------------+----------+--------------+------------------+--------+--------+
| 5          | 1        | CORBA        | /var/tmp/ids1.ior |     1 |      2 |
+------------+----------+--------------+------------------+--------+--------+
```

A **SOAP** instance configuration looks like this:

```
+------------+----------+--------------+------------------+--------+--------+
| instanceID | serverID | instanceName | endpoint         | active | typeID |
+------------+----------+--------------+------------------+--------+--------+
| 6          | 2        | SOAP         | http://host:port/ |     1 |      2 |
+------------+----------+--------------+------------------+--------+--------+
```

An **OFFLINE** instance configuration looks like this:

```
+------------+----------+--------------+------------------+--------+--------+
| instanceID | serverID | instanceName | endpoint         | active | typeID |
+------------+----------+--------------+------------------+--------+--------+
| 5          | 3        | OFFLINE      | 8                |     1 |      2 |
+------------+----------+--------------+------------------+--------+--------+
```

6.4 Whiteboard Installation

You can start the Whiteboard with any of these Internet Browsers

- Firefox version 7 or newer
- Chrome version 15 or newer

In both cases you also need the Shockwave Flex Browser Plug-In Version 11.

Make sure Popups are allowed on the client workstations for the print:cometserver host, otherwise you cannot use the Whiteboard.

Usually you get redirected to the Flex GUI from the priint:suite publication planner. If you want to integrate the Whiteboard in other environments, you will use the Whiteboard Servlet, which can be found at http://your.server.adress/CometServer/Whiteboard. See the Server Development documentation for more information.

Whiteboard Installation

1. unpack the file Whiteboard/PublicationsGUI.zip
2. move the folder just created to the domains DocumentRoot directory, this is on
   - **Windows**: *C:\glassfish\domains\domain1\docroot*
   - **Mac**: */glassfish/domains/domain1/docroot*

6.5 Start / Stop GlassFish Server

6.5.1 Considerations

See Section "5.7.1 Considerations" for InDesignServer, the same applies for GlassFish Server.

In addition, you must restart GlassFish whenever the network configuration changes. This is often the case if you run priint:cometserver on a mobile device.

6.5.2 Manually Start / Stop GlassFish On Microsoft Windows

Open a command prompt, change to the GlassFish installation directory

```
cd C:\glassfish\
```

Start a GlassFish domain (type all in one line)

```
bin\asadmin.bat start-domain
```

This will start the default domain (domain1), if you configured more than one domain, you must provide the domain name also, e.g.

```
bin\asadmin.bat start-domain priint
```

To stop GlassFish, type

```
bin\asadmin.bat stop-domain
```

Again, if there is more than one domain, you must specify the domain, e.g.

```
bin\asadmin.bat stop-domain priint
```

6.5.3 Manually Start / Stop GlassFish On Apple Mac OS X

Open a Terminal window, change to the GlassFish installation directory

```
cd /glassfish/
```

Start a GlassFish domain (type all in one line)

```
./bin/asadmin start-domain
```

This will start the default domain (domain1), if you configured more than one domain, you must provide the domain name also, e.g.

```
./bin/asadmin start-domain priint
```

To stop GlassFish, type

```
./bin/asadmin stop-domain
```

Again, if there is more than one domain, you must specify the domain, e.g.

```
./bin/asadmin stop-domain priint
```

6.5.4 Running GlassFish As A Windows Service

The final release of Suns Java Application Server 2.1.1 shipped with a graphical installer, including the option to install a Windows service. The graphical installer isn't available for download anymore. You can request a copy from Werk II, but since there seems to be no official support from Oracle, you will hardly get any support for this version of the installer at all.

As an alternative, I would suggest using the *sc.exe* command to register and manage Windows services. Since it's sometimes difficult to set up the parameters properly, a java program is included in the installation package, which I highly recommend using:

glassfishsvc.jar

You find this utility in *Resources/Tools/Windows* after unpacking *Resources/Tools.zip*.

The program is written by Ryan de Laplante and based on the original Sun service part of the graphical installer. Parts of it therefore may be covered by the Sun CDDL license; source code is included in the jar file. Basically, the program calls *sc.exe* with properly escaped parameters.

Install GlassFish Windows Service

1. copy *glassfishsvc.jar* to the glassfish installation directory (e.g. *C:\glassfish*)
2. open a command prompt, cd to the GlassFish installation directory and run

   ```
   java -jar glassfishsvc.jar -i
   ```

   This will install a Windows service named *GlassfishAppServer*. The service will start / stop the default domain *domain1*. Use the Windows service control panel to start / stop the service or change service properties.

The *glassfishsvc* utility supports a number of options, such as providing a service name, domain name and GlassFish credentials. You can use the command above to install a Windows service with the default service name (*GlassfishAppServer*) with a default domain (*domain1*). As far as I can see, there should be no reason not to do so. However, to see a list of all options and more examples, simply run

   ```
   java -jar glassfishsvc.jar
   ```

Use the Windows Service Control Panel to start and stop the GlassFish Application Server.

6.5.5 Running GlassFish As A Mac OS X Service using launchd

To start and run GlassFIsh using launchd, you must define a daemon for the GlassFish server. To do so, put a configuration file into /Library/LaunchDaemons/. This file should have a reasonable name (so it can be identified for later editing) and must have the extension *.plist*.

Example

File: /Library/LaunchDaemons/com.sun.glassfish.plist

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE plist PUBLIC ="-//Apple Computer//DTD PLIST 1.0//EN"
"http://www.apple.com/DTDs/PropertyList-1.0.dtd">
<plist version="1.0">
<dict>
  <key>Label</key>
  <string>com.sun.glassfish</string>
  <key>UserName</key>
  <string>priint</string>
  <key>GroupName</key>
  <string>wheel</string>
  <key>OnDemand</key>
  <false/>
  <key>ProgramArguments</key>
  <array>
    <string>/glassfish/bin/gfservice</string>
    <string>start-domain</string>
    <string>domain1</string>
  </array>
  <key>RunAtLoad</key>
  <true/>
 </dict>
</plist>
```

Before using this property list, you might have to adapt the username, group name and path to the gfservice script (*/glassfish/bin/gfservice*). The latter is a script provided by Werk II. Basically this is just a wrapper for the asadmin command, you can edit it with a text editor and adapt it to your needs.

In short, to **configure launchd to start GlassFish**

1. copy the *gfservice* script to your GlassFish bin directory (e.g. */glassfish/bin/*), check, that the script is executable (you might have to "chmod +x" after copying)
2. create a plist file like shown above in */Library/LaunchDaemons*

On system startup all daemons found in /Library/LaunchDaemons are started (unless OnDemand is set to *true*, in this case you would have to bind the daemon to a certain port, so launchd can determine, *when* the service is demanded).

To start the service manually, open a Terminal window and type

```
sudo launchctl load /Library/LaunchDaemons/com.priint.comet1.plist
```

To stop manually, use

```
sudo launchctl unload /Library/LaunchDaemons/com.priint.comet1.plist
```

It is not recommended saving Daemon files in the */System/Library/LaunchDaemons* folder, these entries may be deleted on system updates. Daemons in */Library/LaunchDaemons* will be started literally the same way like daemons in */System/Library/LaunchDaemons*.

7   Data Connectivity

7.1   MySQL / ODBC

7.1.1   General Hints

To connect to a MySQL database with *InDesign Desktop* or *InDesignServer*, you need an ODBC driver for MySQL. We recommend and support the **mysql-connector-odbc-5.1.11** driver.

Drivers for all supported platforms can be downloaded from the MySQL download site

http://www.mysql.com/downloads/connector/odbc/

When connecting with ODBC, you first have to configure a *data source name* (DSN), this will be referred as the "Service" in the Plug-In login dialogue or connection configuration file.

To connect to a MySQL database with priint:cometserver no additional installation is required (the mysql JDBC drivers are included in the *3dpartylibs* archive, which hopefully you have installed already).

7.1.2   Configuring MySQL Connector/ODBC On Microsoft Windows

Download and install the driver suitable for your OS and architecture.

- download a **32bit** driver for InDesign Desktop or InDesignServer 32bit
- download a **64bit** driver for InDesignServer 64bit

To configure datasource names you have to launch the *Windows ODBC Datasource Panel.* Windows x64 comes shipped with two versions of this panel, one for 32bit drivers and one for 64bit.

- **32bit:** launch the

    `C:\Windows\SysWOW64\odbcad32.exe`

    Datasource panel for InDesign *Desktop* or InDesignServer *32bit*
- **64bit**: launch the

    `C:\Windows\System32\odbcad32.exe`

    Datasource panel for InDesignServer *64bit*

To be sure to use the right application, we recommend launching the Datasource panel from an Explorer window rather than from the System Administration panel.

And, yes. The information above *is* correct. To fix broken ODBC configuration can be quite a pain, especially if you uninstalled or updated drivers in the meantime.

If you run into *any* trouble related with odbc database connections, please

- remove the ODBC datasource definition **completely**. If you are not able to remove it from the *Windows ODBC Datasource Panel*, search the *Windows registry* for the datasource name and remove all related keys.
- remove the ODBC driver **completely**
- reinstall the driver and reconfigure your DSN
- restart all applications using this DSN

Set up a new data source

1. launch the *ODBC Datasource panel* as described above

2. add a new System DSN

3. select *MySQL ODBC 5.1 Driver* in the list of available drivers

4. in the setup dialogue enter the following values

   ▪ **Data Source Name:** demo, **Description**: anything descriptive you like
   ▪ **Server**: IP address or name of the database server, **Port**: usually 3306
   ▪ **User**: demo, **Password**: priintdemo
   ▪ **Database**: comet_config

   These values apply for the installation default settings, they may differ for your installation.

5. Click the *Details* button, in the *Connection* tab activate the following settings:

   ▪ *Allow big result sets, Enable automatic reconnect, Allow multiple statements*
   ▪ Select the **Character Set** utf8

6. in the Misc tab activate Ignore space after function names

7. test the connection using the *Test* button, finally click *OK* to save the new DSN

7.1.3 Configuring MySQL Connector/ODBC On Apple Mac OS X

Download and install the driver suitable for your OS version:

- to use the driver with InDesign or InDesignServer, **you must install the 32bit version**
- for Mac OS X 10.7 and 10.8 you can use the 10.6 installation package.

It seems that MySQL has dropped development of setup libraries from version 10.6. That means, using the ODBC Administrator applications (either */Application/Utilities/OpenLink ODBC Administrator.app* or */Applications/iODBC/iODBC Administrator.app*) can only provide a standard setup interface fort he MySQL ODBC driver. You can use this interface, but, considering that both administration panels tend to crash quite often, I find it more comfortable to edit the odbc configuration files manually.

Manually setup ODBC connection

1. if you want to test the ODBC installation (you do not have to edit this file):
   open */Library/ODBC/odbcinst.ini* with a text editor. If the driver has installed properly, you should find the following lines:

```
[ODBC Drivers]
MySQL ODBC 5.1 Driver                                = Installed

[MySQL ODBC 5.1 Driver]
Driver = /usr/local/lib/libmyodbc5.so
```

2. open */Library/ODBC/odbc.ini* with a Text editor, you may need super user privileges to do that

3. in the *[ODBC Data Sources]* section add the line

```
[ODBC Data Sources]
demo = MySQL ODBC 5.1 Driver
```

4. at the end oft he document add the following lines

```
[demo]
Driver          = /usr/local/lib/libmyodbc5.so
Description     = demo database
CHARSET         = utf8
DATABASE        = comet_config
AUTO RECONNECT  = 1
BIG_PACKETS     = 1
```

```
IGNORE_SPACE     = 1
MUTLI_STATEMENTS = 1
PORT             = 3306
SERVER           = database.server.address
```

5. save the file. To test the connection launch one of the ODBC Administrator applications, select the *System DSN* tab, select the new entry in the list of DSNs and click the *Test* button.

6. to set up a DSN with a different name or another DSN, you would add the following to the */Library/ODBC/odbc.ini* file:

```
[ODBC Data Source]
othername = MySQL ODBC 5.1 Driver

[othername]
Driver           = /usr/local/lib/libmyodbc5.so
etc. etc.
```

7.1.4 Connect InDesign Desktop With MySQL

Install and configure the MySQL Connector/ODBC drivers like explained in section "7.1.2 Configuring MySQL Connector/ODBC On Microsoft Windows" or "7.1.3 Configuring MySQL Connector/ODBC On Apple Mac OS X".

In the InDesign database login dialogue enter the following values

*   **Service**: name of the data source (e.g. *demo*)
*   **User**: database user / account (e.g. *demo*)
*   **Password**: account password (e.g. *priintdemo*)
*   **Database**: you can leave this field empty, if you provided the database name in the DSN configuration, otherwise the name of the database (e.g. *comet_config*)
*   **Charset**: UTF8
*   **Client**: you can leave this field empty

7.1.5 Connect InDesignServer With MySQL

Install and configure the MySQL Connector/ODBC drivers like explained in section "7.1.2 Configuring MySQL Connector/ODBC On Microsoft Windows" or "7.1.3 Configuring MySQL Connector/ODBC On Apple Mac OS X".

The priint:comet Server Plug-Ins read connection data from a XML file. In a priint:cometserver environment this is the *connection.xml* file in the CometEnvironment folder (which you hopefully copied to the right destination).

```
<!-- File: CometEnvironment/connections.xml -->
<connections>
  <connection>
    <id>1</id>                          <!-- must be unique -->

    <!--
       this name must match the name of the corresponding
       priint:cometserver dataset
    -->
    <name>demo</name>

    <type>odbc_utf8</type>              <!-- must be odbc utf8 for MySQL -->
    <server>demo</server>               <!-- name oft he data source -->
```

```
      <user>demo</user>
      <password>priintdemo</password>
      <db></db>
      <lang></lang>                       <!-- SOAP only -->
      <client></client>
      <passcredentials></passcredentials> <!-- currently unused -->
   </connection>
</connections>
```

! If you

- change any of the connection configuration in priint:cometserver / GlassFish
- use different database names or database accounts or
- use a different Datasource name in the ODBC configuration

you have to change the connection entry in *connections.xml* too. After changing the file you must restart all InDesign server instances using this configuration.

7.1.6 Connect priint:cometserver With MySQL

With GlassFish we use a connection pool to connect with databases. The second step will be to bind this connection pool to a named JDBC resource, this is documented in Section "6.3.2 Configure JDBC Resources".

During a typical priint:cometserver installation you will have to do the following at least twice, because we need to connect to the *comet_admin* and to the *comet_config* database.

To configure a connection pool

1. GlassFish Server must be running
2. login to the GlassFish administration console (http://localhost:4848 or http://ip.address:4848)
3. navigate to *Resources > JDBC > Connection Pools*, click *New…* in the right panel. Configure the Connection Pool with the following settings:
   - **Name**: *comet_admin* (for comet_config: *comet_config*)
   - **Resource Type:** select javax.sql.ConnectionPoolDataSource
   - **Database Vendor:** select MySQL
4. click the *Next* button to proceed to further settings
5. the next page allows quite a few settings, so section by section
- **General Settings:** all settings can be left unchanged
- **Pool Settings**: by now the default settings can be left unchanged.
- **Connection Validation**: activate the following settings
   - *Connection Validation: Required*
   - *On Any Failure: Close All Connections*
   - *Allow Non Component Callers: Enabled*
- **Transaction**: activate the following settings
   - *Non Transactional Connections: Enabled*
   - *Transaction Isolation:* select *read-committed*
   - *Isolation Level: Guaranteed*
- **Additional Properties**

- **Password**: *priintadmin* (for comet_config: *priintdemo*)
- **Url / URL**: you have to provide both parameters (though with same values)!
  jdbc:mysql://localhost:3306/comet_admin
  (or *jdbc:mysql://hostname.or.ip:3306/comet_admin*, if the database server runs on a different host, for comet_config the URL is *jdbc:mysql://hostname.or.ip:3306/comet_config*)
- **User**: *admin* (for comet_config: *demo*)

6. click *Finish* at the top or at the bottom of the page

7. in the connection pool list you should now see the new entry, click on the name and test the connection using the *Ping* button at the top of the page.

7.2 Microsoft SQL Server / ODBC

7.2.1 General Hints

To connect to a Microsoft SQL Server database with *InDesign Desktop* or *InDesignServer*, you need an ODBC driver for SQL Server. See the sections below to learn how to install and configure a suitable driver for your operating system.

When connecting with ODBC, you first have to configure a *data source name* (DSN), this will be referred as the "Service" in the Plug-In login dialogue or connection configuration file.

To connect to a SQL Server database with priint:cometserver no additional installation is required (the required JDBC drivers are included in the *3dpartylibs* archive, which hopefully you have installed already).

7.2.2 Configuring The SQL Native ODBC Driver On Microsoft Windows

On Windows you need the *Microsoft SQL Server 2008 Native Client* driver to connect with a SQL Server database. The recommended and supported version is *SQL Server Native Client 10.0*. We recommend checking the Microsoft download site for the latest release:

http://www.microsoft.com/en-us/download/details.aspx?id=3522

The SQL Server Native Client 10.0 is part of the SQL Server 2008 feature pack, which is available for virtually any Windows platform. On the download page search for "*ODBC*" and select the installer suitable for your OS.

Check driver installation

1. launch the *ODBC Datasource Panel*
2. select the *Driver* tab
3. check, if *SQL Server Native Client 10.0* appears in the list of available drivers
4. it's not the *SQL Native Client* or *SQL Server* driver, these drivers may work in your environment, but are not supported by Werk II
5. if the driver does not appear in the list, download and install it as stated above

Set up a new data source

1. launch the *ODBC Datasource Panel*
2. add a new System DSN
3. select *SQL Server Native Client 10.0* in the list of available drivers (again: do not select the *SQL Server* or just *SQL Native Client* driver)
4. in the setup dialogue enter the following values
   ▪ **Name:** demo
   ▪ **Description**: anything descriptive you like
   ▪ **Server**: this can be either the IP address / hostname and port of the database server (e.g. my.sqlserver.db:1433) or an instance name (such as *HOST-NAME\MSSQLSERVER*)
   Click *Next* to proceed to further settings.
5. choose *SQL Server authentication* on the next screen and enter valid credentials (login ID and password). Make sure the "connect …" checkbox at the bottom of the panel is activated and

click *Next* to proceed to the next screen

6. if the connection data and credentials provided are correct, you should be presented with a list of available databases. Select the standard database, in our case *comet_config*.
    Leave the other settings unchanged and click *Next* to proceed to the next screen

7. activate the *Convert charset data* checkbox. All other settings can be left unchanged.

8. click *Finish* and test the connection

7.2.3 Configuring The Openlink ODBC Driver On Apple Mac OS X

The only drivers known to work properly with Mac OS X are the Openlink Single-Tier Express and Lite drivers version 6.1. Downloads are available at

<div align="center">

http://uda.openlinksw.com/odbc-sqlserver-ee/ or

http://uda.openlinksw.com/odbc-sqlserver-st/

</div>

You need to register at OpenLink to download these drivers, also a license is required. More information can be obtained from the OpenLink website.

Download and install the driver according to the OpenLink documentation.

To setup a new datasource

1. launch the *ODBC Administrator* application (either */Application/Utilities/OpenLink ODBC Administrator.app* or */Applications/iODBC/iODBC Administrator.app*)

2. add a new System DSN

3. select OpenLink *SQLServer Driver (Express Edition)***(Unicode)** in the list of available drivers (do not select the Non-Unicode driver)

4. in the setup dialogue enter the following values
    ▪ **DSN:** demo
    ▪ **Description**: anything descriptive you like
    Click *Continue* to proceed to further settings.

5. configure the connection using the following values
    ▪ **Server Type**: MSSQL
    ▪ **Host**: database host (IP or host name)
    ▪ **Port**: database port, default is 1433
    ▪ **Database**: standard database, e.g. *comet_config*
    ▪ **User name**: database user, e.g. *demo*
    The *Advanced* settings can be left unchanged. Click Continue to proceed to the next screen.

6. settings in the *Options* screen can be left unchanged. Click *Continue* to proceed to the next screen.

7. activate *Multiple Active Statements Emulation* on the Preferences screen. Click Continue to proceed.

8. click the *Test Data Source…* button and finally *Finish*

7.2.4   Connect InDesign Desktop With SQL Server

Configure a suitable ODBC driver for your system like explained in "7.2.2 Configuring The SQL Native ODBC Driver On Microsoft Windows" or "7.2.3 Configuring The Openlink ODBC Driver On Apple Mac OS X".

In the InDesign database login dialogue enter the following values

- **Service**: name of the data source (e.g. *demo*)
- **User**: database user / account (e.g. *demo*)
- **Password**: account password (e.g. *priintdemo*)
- **Database**: you can leave this field empty, if you provided the database name in the DSN configuration, otherwise the name of the database (e.g. *comet_config*)
- **Charset**: Unicode
- **Client**: you can leave this field empty.

7.2.5   Connect InDesignServer With SQL Server

Configure a suitable ODBC driver for your system like explained in "7.2.2 Configuring The SQL Native ODBC Driver On Microsoft Windows" or "7.2.3 Configuring The Openlink ODBC Driver On Apple Mac OS X".

The priint:comet Server Plug-Ins read connection data from a XML file. In a priint:cometserver environment this is the *connection.xml* file in the CometEnvironment folder (which you hopefully copied to the right destination).

```
<!-- File: CometEnvironment/connections.xml -->
<connections>
  <connection>
    <id>1</id>                         <!-- must be unique -->

    <!--
       this name must match the name of the corresponding
       priint:cometserver dataset
    -->
    <name>demo</name>

    <type>odbc_unicode</type>          <!-- must be odbc_unicode for
                                               Microsoft SQL Server -->
    <server>demo</server>              <!-- name oft he data source -->
    <user>demo</user>
    <password>priintdemo</password>
    <db></db>
    <lang></lang>                      <!-- SOAP only -->
    <client></client>
    <passcredentials></passcredentials>  <!-- currently unused -->
  </connection>
</connections>
```

If you

- change any of the connection configuration in priint:cometserver / GlassFish
- use different database names or database accounts or
- use a different Datasource name in the ODBC configuration

you have to change the connection entry in *connections.xml* too. After changing the file you must restart all InDesign server instances using this configuration.

7.2.6 Connect priint:cometserver With SQL Server

With GlassFish we use a connection pool to connect with databases. The second step will be to bind this connection pool to a named JDBC resource, this is documented in Section "6.3.2 Configure JDBC Resources".

During a typical priint:cometserver installation you will have to do the following at least twice, because we need to connect to the *comet_admin* and to the *comet_config* database.

To configure a connection pool

1. GlassFish Server must be running
2. login to the GlassFish administration console (http://localhost:4848 or http://ip.address:4848)
3. navigate to *Resources > JDBC > Connection Pools*, click *New…* in the right panel. Configure the Connection Pool with the following settings:
    - **Name**: *comet_admin* (for comet_config: *comet_config*)
    - **Resource Type:** select *javax.sql.ConnectionPoolDataSource*
    - **Database Vendor:** select *Microsoft SQL Server*
4. click the *Next* button to proceed to further settings
5. the next page allows quite a few settings, so section by section
- **General Settings:**
    - *Datasource Classname*
      com.microsoft.sqlserver.jdbc.SQLServerConnectionPoolDataSource
- **Pool Settings**: by now the default settings can be left unchanged.
- **Connection Validation**: activate the following settings
    - *Connection Validation: Required*
    - *On Any Failure: Close All Connections*
    - *Allow Non Component Callers: Enabled*
- **Transaction**: activate the following settings
    - *Non Transactional Connections: Enabled*
    - *Transaction Isolation:* select *read-committed*
    - *Isolation Level: Guaranteed*
- **Additional Properties**
    - **databaseName**: *comet_admin* (for comet_config: *comet_config*)
    - **networkProtocol**: *tcp*
    - **password**: *priintadmin* (for comet_config: *priintdemo*)
    - **portNumber**: 1433
    - **serverName**: IP or hostname of the database server
    - **User**: *admin* (for comet_config: *demo*)
- add the following property using the *Add Property* button
    - **sendStringParametersAsUnicode**: *false*
6. click *Finish* at the top or at the bottom of the page, test the connection by choosing the new entry from the list of configured pools and clicking the *Ping* button at the top of the page.

7.3 Oracle / OCI

7.3.1 General Hints

To connect to an Oracle database server with *InDesign Desktop* or *InDesignServer*, you need the *OCI* version of the CoreService Plug-Ins (either *CoreService[Oracle]* or *CoreService[Server][Oracle]*) and the *Oracle Instant Client* aka *OCI* libraries.

TNS Listener must be enabled on the database server side and you must configure some environment variables and (TNS) service names.

To connect to an Oracle database server with priint:cometserver no additional installation is required (the required JDBC drivers are included in the *3dpartylibs* archive, which hopefully you have installed already).

7.3.2 Configuring OCI On Microsoft Windows

You need Oracle Instant Client version 10.2.0.4, which can be downloaded from the Oracle download site. To install and use OCI follow these instructions:

1. download and unzip the package from the Oracle download site, you need the file

    instantclient-basic-win32-10.2.0.4.zip

    Note: download the **32bit** version

2. follow the installation instructions from the Oracle website.

3. launch System Properties > Advanced and click "Environment Variables", add the following environment variables

    ▪ **NLS_LANG**: GERMAN_GERMANY.AL32UTF8

    (it **must** be AL32UTF8, but you may want to set another locale)

4. **TNS_ADMIN**: path to directory containing the *tnsnames.ora* file, e.g. *C:\oracle*

5. **TNS Names**

    create the *tnsnames.ora* file in the directory provided above, add an entry for the connection to your comet_config database, e.g

    ```
    DEMO=(DESCRIPTION=
    (ADDRESS_LIST=
     (ADDRESS=
       (PROTOCOL=TCP)
       (HOST=192.168.56.101)
       (PORT=1521) ) )
    (CONNECT_DATA=
     (SID=DEMO) ) )
    ```

    Strangely it is important to create / edit the file on a *Windows* system. Line ending encoding must match the default encoding of the target system.

7.3.3 Configuring OCI On Apple Mac OS X

1. download and unzip the package from the Oracle download site, you need the file

    instantclient-basiclite-10.2.0.4.0-macosx-x86.zip

    Note: download the **32bit** version

2. do not bother the installation instructions from the Oracle website, instead

3. in a Terminal window run the following command

```
sudo mv ~/Downloads/instantclient_10_2/*.dylib* /usr/lib/
```

4. **Oracle Environment On Mac OSX**

   Mac OSX 10.5 - 10.7

   In former versions of Mac OSX (including 10.7) it was possible to set environment variables in the *environment.plist* file located in *~/.MacOSX.*

   - create the folder *.MacOSX* in your home directory (if it doesn't already exist)
   - create the file *environment.plist* in the *.MacOSX* folder
   - open the file in a text editor and paste the following lines:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE plist PUBLIC ="-//Apple Computer//DTD PLIST 1.0//EN"
"http://www.apple.com/DTDs/PropertyList-1.0.dtd">
<plist version="1.0">
    <dict>
        <key>TNS ADMIN</key>
        <string>/Library/Preferences/Oracle/</string>
        <key>NLS_LANG</key>
        <string>GERMAN GERMANY.AL32UTF8</string>
    </dict>
</plist>
```

   - alternatively you can use the *Apple Property List Editor* application to edit the file

   Mac OSX 10.8

   On Mac OSX 10.8 you have to set the Oracle environment in the */etc/launchd.conf* file:

   - open the file /etc/launchd.conf with superuser privileges in a text editor, e.g.

```
sudo vim /etc/launchd.conf
```

   - add the following lines and save the file:

```
# It must be AL32UTF8, but you may want to choose another locale:
setenv NLS_LANG GERMAN_GERMANY.AL32UTF8
 # This would be a proper place, though you can keep the
 # tnsnames.ora file anywhere else:
setenv TNS_ADMIN /Library/Preferences/Oracle/
```

   - logout from Mac OSX and login again, so the environment variables will be read from launchd.conf

5. **TNS Names**

   create the *tnsnames.ora* file in the directory provided above, add an entry for the connection to your comet_config database, e.g.

```
DEMO=(DESCRIPTION=
(ADDRESS_LIST=
 (ADDRESS=
   (PROTOCOL=TCP)
   (HOST=192.168.56.101)
   (PORT=1521) ) )
(CONNECT_DATA=
 (SID=DEMO) ) )
```

   Strangely it is important to create / edit the file on a *Mac* system. Line ending encoding must match the default encoding of the target system.

7.3.4 Connect InDesign Desktop With Oracle

As mentioned above, you must install the CoreService[Oracle] Plug-In. This replaces the CoreService[Database] or just CoreService Plug-In installed for other database connections.

Configure the OCI client and a TNS name like explained in "7.3.2 Configuring OCI On Microsoft Windows" or "7.3.3 Configuring OCI On Apple Mac OS X"

In the InDesign database login dialogue enter the following values

- **Service**: TNS name of the data source (e.g. *demo*)
- **User**: database user / account (e.g. *demo*)
- **Password**: account password (e.g. *priintdemo*)
- **Database**: you can leave this field empty.
- **Charset**: UTF-8
- **Client**: you can leave this field empty. If provided, only placeholders, templates etc. of the given client (i.e. domain) will be shown.

7.3.5   Connect InDesignServer With Oracle

As mentioned above, you must install the *CoreService [Oracle][Server]* Plug-In. This replaces the *CoreService [Server]* Plug-In installed for other database connections.

Configure the OCI client and a TNS name like explained in "7.3.2 Configuring OCI On Microsoft Windows" or "7.3.3 Configuring OCI On Apple Mac OS X"

The priint:comet Server Plug-Ins read connection data from a XML file. In a priint:cometserver environment this is the *connection.xml* file in the CometEnvironment folder (which you hopefully copied to the right destination).

```
<!-- File: CometEnvironment/connections.xml -->
<connections>
  <connection>
    <id>1</id>                          <!-- must be unique -->

    <!--
       this name must match the name of the corresponding
       priint:cometserver dataset
    -->
    <name>demo</name>

    <type>oracle_utf8</type>            <!-- must be oracle utf8 for
                                                Oracle Database Server -->
    <server>demo</server>              <!-- name oft he data source -->
    <user>demo</user>
    <password>priintdemo</password>
    <db></db>
    <lang></lang>                       <!-- SOAP only -->
    <client></client>
    <passcredentials></passcredentials>  <!-- currently unused -->
  </connection>
</connections>
```

!   If you

- change any of the connection configuration in priint:cometserver / GlassFish
- use different database names or database accounts or
- use a different TNS name

you have to change the connection entry in *connections.xml* too. After changing the file you must restart all InDesign server instances using this configuration.

7.3.6 Connect priint:cometserver With Oracle

With GlassFish we use a connection pool to connect with databases. The second step will be to bind this connection pool to a named JDBC resource, this is documented in Section "6.3.2 Configure JDBC Resources".

During a typical priint:cometserver installation you will have to do the following at least twice, because we need to connect to the *comet_admin* and to the *comet_config* database.

To configure a connection pool

1. GlassFish Server must be running
2. login to the GlassFish administration console (http://localhost:4848 or http://ip.address:4848)
3. navigate to *Resources > JDBC > Connection Pools*, click *New…* in the right panel. Configure the Connection Pool with the following settings:
    - **Name**: *comet_admin* (for comet_config: *comet_config*)
    - **Resource Type:** select *javax.sql.ConnectionPoolDataSource*
    - **Database Vendor:** select *Oracle*
4. click the *Next* button to proceed to further settings
5. the next page allows quite a few settings, so section by section
- **General Settings:**
    - *Datasource Classname*
      oracle.jdbc.pool.OracleConnectionPoolDataSource
- **Pool Settings**: by now the default settings can be left unchanged.
- **Connection Validation**: activate the following settings
    - *Connection Validation: Required*
    - *On Any Failure: Close All Connections*
    - *Allow Non Component Callers: Enabled*
- **Transaction**: activate the following settings
    - *Non Transactional Connections: Enabled*
    - *Transaction Isolation:* select *read-committed*
    - *Isolation Level: Guaranteed*
- **Additional Properties**
    - depending on the connection type, provide the required information (e.g. TNS name or SID) in the property list below
6. click *Finish* at the top or at the bottom of the page
7. in the connection pool list you should now see the new entry, click on the name and test the connection using the *Ping* button at the top of the page.

7.4 SOAP WebService

Documentation on how to setup SOAP WebService connections will follow in a future version of this guide. In the meantime, please refer to the Plug-Ins documentation or support@priint.net.

7.4.1 General Hints

7.4.2 Connect InDesign Desktop With WebServices

7.4.3 Connect InDesignServer With WebServices

7.4.4 Connect priint:cometserver With WebServices

7.5 XML Offline

Documentation on how to setup XML Offline connections will follow in a future version of this guide. In the meantime, please refer to the Plug-Ins documentation or support@priint.net.

7.5.1 General Hints

7.5.2 Connect InDesign Desktop With XML Offline

7.5.3 Connect InDesignServer With XML Offline

7.5.4 Connect priint:cometserver With XML Offline

This is not possible. XML Offline is mainly intended for small projects and cannot be used in a priint:cometserver context.

A. priint:comet Plug-Ins

A.1 priint:comet Desktop Plug-Ins

Install legend: **X** – required | **O** - recommended | **-** - optional

Bold Plug-In names indicate a typical InDesign workstation setup for general purpose. If in doubt, we recommend installing these Plug-Ins.

Plug-In / File Name	Description	Comment	Install
Comet	Comet Reader	Only one of these Plug-Ins must be present	X
CometXML	Scripting Support / Table Module		
Comet [ePaper]	Obsolete, special purpose only		
Comet++.	Obsolete, special purpose only		
CoreSelection	Selection support, required!		X
CoreService	All connection types (except of OCI)	Only one of these Plug-Ins must be present	X
CoreService [Database]	Database / ODBC connections only		
CoreService [Internet]	SOAP connections only		
CoreService [Oracle]	Oracle / OCI connections only		
CoreService [XML]	XML Offline only		
CoreServlett	Comet Reader		
DataFiles	File mappings / global script variables		O
Designate	Designate toolbox		O
Documents	Publication panel		O
EmbeddedLink	Support for text placeholders and tags		X
Exportgruppen	Work presets, special purpose only		-
FrameLink	Support for frame placeholders and tags		X
FrameRules	Frame and layout rules panel		O
PageAdapter	Non-proportional adaption full version	Only one of these Plug-In sets (UI / Model) must be present	X
PageAdapter [Model]			
PageAdapter-Reader	Non-proportional adaption build support and reader		
PageAdapter-Reader [Model]			
PageGrids	Grid panel and grid wizard		O
PageItems	Templates panel and features		O
PageTemplates [Model]	Page templates panel and features		X
PageTemplates			
Placeholder	Placeholder panel		X
PlaceholderValues	Placeholder information panel		O
PlainCopy	Copy & Paste plain content		O
Previews	Alternative content and product snippets		O
Products	Product panel and build support		O
Sudoku	Sudoku		-
ToDoList	Comments and Tasks		O
ToDoList [Model]			
XCell	Link Excel spreadsheets		-

A.2 priint:comet Server Plug-Ins

Install legend: **X** – required | **O** - recommended | **-** - optional

Bold Plug-In names indicate a typical InDesignServer setup for general purpose. If in doubt, we recommend installing these Plug-Ins.

Plug-In / File Name	Description	Comment	Install
Comet [Server]	Scripting Support		**X**
CoreService [Server]	All connection types (except of OCI)	Only one of these Plug-Ins must be present	**X**
CoreService [Internet][Server]	SOAP connections only		
CoreService [Oracle] [Server]	Oracle / OCI connections only		
CoreServlett[Server]	Comet Reader		
DataFiles[Server]	File mappings / global script variables		**O**
EmbeddedLink[Server]	Support for text placeholders and tags		**X**
FrameLink[Server]	Support for frame placeholders and tags		**X**
PageAdapter[Server]	Non-proportional adaption full version	Only one of these Plug-In sets (UI / Model) must be present	**X**
PageAdapter [Model][Server]			
PageAdapter-Reader[Server]	Non-proportional adaption build support and reader		
PageAdapter-Reader [Model] [Server]			
PageTemplates [Model]	Page templates features		**X**
PageTemplates			
Placeholder	Placeholder support		**X**
Products	Product build support		**O**
ToDoList	Comments		**O**
ToDoList [Model]			

priint:comet Server Connector Plug-Ins

Plug-In / File Name	Description	Comment	Install
Corba Support	Connect with priint:cometserver using CORBA	Only one of these Plug-Ins should be present	**X**
Soap Support	Connect with priint:cometserver using SOAP		

! **Corba Support:** Starting with CS6 / v 3.3.1 Corba isn't supported any more. The Soap Support Plug-In is just packed with the normal server Plug-Ins, so there is no extra connector Plug-Ins package.

64bit on Windows: if you run InDesignServer x64, you must not use Plug-Ins prior to Revision R3490.

B. InDesignServer Command Line Arguments

The following arguments can be passed to InDesignServer on startup. Example (Windows):

```
InDesignServer.com –argumentname [argumentvalue]
```

Some arguments do and some don't require a value. Example (Mac OS X):

```
./InDesignServer -configuration ids1 -previews -iorfile /var/ids1.ior
```

Keep in mind, that the shell will expand values; this applies for wild cards, variables, environment etc. The following example will give you an idea of how to make use of this:

```
./InDesignServer –iorfile $COMET_TMP/ids1.ior –pluginpath ~/myplugins/
```

B.1 Adobe InDesignServer Command Line Arguments

Please refer to the Adobe InDesignServer documentation for a complete list of standard arguments. The list below only includes arguments mandatory or useful for priint:comet applications.

Argument Name	Argument Value	Description
configuration	<configuration name>	If you run several instances on one InDesignServer host, each instance must have an unique "configuration name" (such as "instance1", "instance2" etc.)
iorfile	<path to iorfile>	This is the communication endpoint for Corba connections. You must provide a file name, the destination folder must exist and must be writeable for the InDesignServer user. If the file provided already exists, this file must be writeable for the InDesignServer user.
previews	-	Calculate previews for embedded or linked images. For whiteboard applications or if documents are exchanged with InDesign Desktop you should always turn this option on, otherwise you will see grey squares instead of images in previews and when first opening the document with InDesign Desktop
nopreviews	-	Turn preview calculation of
LogToApplicationEventLog	-	Redirect output to the system console (Mac) or Event Log (Windows)
pluginpath	<path[,path]>	Additional plug-in folder paths, relative to the application. This can be useful in multi instance environments when running several instances with different purposes (e.g. with / without priint:comet Server Plug-Ins)

B.2 priint:comet Server Command Line Arguments

Beside shell expansion, the priint:comet arguments also support some cscript variable names for path expansion (like $DOCUMENTS, $PLUGINS etc.). Refer to the cscript documentation for details.

Argument Name	Argument Value	Description
cometconfig	<config.xml path>	Path to the Autocomet configuration file, see Section "5.5 Running InDesignServer In Batch Mode (Autocomet)" for details. To make Plug-In updates easier, we recommend using this option and keep the configuration file outside of the Plug-Ins folder.
cometlog	<comet log file path>	Enable common logging and write logs to <comet log file path>. You should enable logging on development systems and to trace errors on productive systems. The destination folder must exist, otherwise logging will fail.
cometapilog	<api log file path>	Enable API logging and write logs to <api log file path>. This will trace the complete communication between priint:cometserver and InDesignServer. The destination folder must exist, otherwise logging will fail.
cometlogrotate	<# bytes>	Rotate comet logs, if the log file exceeds the size of <# bytes> bytes. Default is 2MB, -1 will disable log rotate at all (not recommended). This setting applies to all log files.
cometlic	<license file path>	Read license from <license file path> rather than w2.lic in the Plug-Ins folder. To make Plug-In updates easier, we recommend using this option and keep the license file outside of the Plug-Ins folder.
cometcache	<cache folder path>	Set the comet cache path. The cache path must be unique for each InDesignServer instance running on the same machine, so it's highly recommended using this option. The default is ~/Documents/XCache, which might not be useful in a server environment.
cometport	<port>	(SOAP Support only) Listen for SOAP requests on the given port. If the Soap Support Plug-In is installed, setting this option will enable the SOAP Service.

cometdrawidle	<true | false>	Enable / disable draw idle features. If turned on (default), InDesignServer will prepare previews whenever "idle". You should not turn it off, except you notice problems with previews.
cometthreshold	<# bytes>	Threshold in bytes before writing script results (like XML strings) in tmp files rather than returning the string. This setting is relevant for the Corba connection only, the default value is 32768. Higher values may improve performance, on the other hand large result strings (> 512k) sometimes cause failures in the Corba connection.
shutdowntrigger **purgetrigger** (from R3508)	one of the following criteria: uptime=nd | nh | nm | ns documents=n jobs=n memory[=low] time=$hh:mm:ss$ headroom=nG | nM | nK	Shutdown the InDesignServer instance or purge memory after (every) n days / hours / minutes / seconds uptime, after n documents or n jobs have been processed, when memory is low, at a certain time or when no more n GB memory can be allocated. See Section "5.8 Advanced InDesignServer Configuration" for more information.

There are several more options related to the SOAP communication, which should be more or less self-explanatory. Changing them is only required, if you encounter problems with the SOAP communication:

cometaccepttimeout	<accept timeout in ms>	default: 1000 (= 1s)
cometreceivetimeout	<receive timeout in ms>	default: 60000 (=60s)
cometsendtimeout	<send timeout in ms>	default: 60000 (=60s)
cometyield	<yield ms>	milliseconds to yield between listen attempts, currently unused

C. Installation Resources

#	Filename	Description
1	ReleaseNotes.txt	Release history
2	InstallationGuide.pdf	This guide
3	Developer/Documentation.zip	Developer documentation
4	Developer/priintwsclient.jar	JAX WS client to integrate priint:cometserver in Java Applications
5	Resources/Images.zip	Images for the standard showcase
6	Resources/Install-DB.zip	Database installation scripts
7	Resources/Masterdocuments.zip	Master documents for the standard showcase, currently CS5 only
8	Resources/Tools.zip	Various tools, such as startup scripts, service registration etc.
9	Server/3dpartylibs.zip	3d party Java libraries required to run priint:cometserver
10	Server/CometEnvironment.zip	Environment settings for InDesignServer instances connected with priint:cometserver
11	Server/cometserver.lic	Default license (all modules enabled)
12	Server/CometServerLight.ear	The priint:cometserver Enterprise Application
13	User-Guides/priint-comet-whiteboard-3-2-2_usermanual_v07_en	Whiteboard User Manual
14	Whiteboard/PublicationsGUI.zip	Whiteboard Flex GUI
15	Workbench/priint.comet.workbench_3.3_2011-11-11_IntelOnly_Setup.zip	Workbench Installer for Apple Mac OS X
16	Workbench/priint.comet.workbench_3.3_2011-11-11_Setup.exe	Workbench Installer for Windows

D. Showcase Installation

To use the showcase (i.e.: the *Product Data* database component) delivered with priint:comet, you have to do two more installation step. You need the following files:

- *Resources/Images.zip*: images for some of showcase products
- *Resources/Masterdocuments.zip*: InDesign CS5 master documents

1. unpack *Resources/Images.zip*, move the *Images* folder to your default *Documents* folder:
   - **Windows**: C:\glassfish\domains\domain1\config\Documents
   - **Mac**: /glassfish/domains/domain1/config/Documents
2. unpack *Resources/Masterdocuments.zip*. This archive contains InDesign documents suitable for the showcase

Note: the showcase data contains templates and master documents for InDesign **CS5**. They *can* be used with other CS versions (CS5.5 or CS6), however you are encouraged to convert them to the CS version you are using.

Using templates in the right CS version generally improves performance and stability.

- you *can* use templates of an older CS version, although this is **not recommended**.
- you *cannot* use master documents of an older CS version with priint:cometserver / Publication Planner

We have prepared the conversion steps for you, so all you have to do is

1. launch InDesign Desktop
2. login to your demo datasource (e.g. service *demo* with user / password *demo* / *priintdemo*)
3. open the *Templates* panel
4. select *Convert All Templates* from the panel menu.
   This will open all templates found in your *comet_config* database and write them back to the database in the new CS version.
5. select *Convert Documents* from the panel menu, choose the folder containing your master documents and select a target folder.
   This will save copies of all master documents in the new CS version in a new folder.

E. Error Codes

E.1 priint:cometserver Exceptions

INVALID_SESSION (1200)

Description: the session ID you used for an API request is not valid or session has expired.

Solution: Discard session and login again.

ACCESS_DENIED (1201)

Description: your account does not privilege you to access the requested ressource.

Solution: there may be many reasons for this exception. One common fault is that the user tries to register a document outside of known repositories or you try to gather information from the server which is only available for administrators (like the list and status of all InDesignServer instances). Most likely this can be fixed by adding missing configuration or using a more privileged account.

ENDPOINT_NOT_FOUND (1223)

Description: this will be thrown if the communication endpoint of an InDesignServer instance (either URL or path to ior file) can not be found or resolved.

Solution: check the InDesign Server configuration in your admin database, check the InDesign Server statuses.

ILLEGAL_ENDPOINT_STATE (1224)

Description: an InDesign Server is in an illegal state

Solution: something went seriously wrong if you catch this exception. Most likely this was caused by a previous api / InDesignServer request, so dont bother trying to debug the piece of code where the exception occured. Check the InDesignServer status, PlugIn logs and API logs.

NO_IDS_INSTANCE_AVAILABLE (1225)

Description: there is no InDesignServer instance available. Either none has been configured in your admin database or all instances are down / unreachable at the moment.

Solution: check InDesignServer configuration and statuses

UNSUPPORTED_PROTOCOL (1226)

Description: the protocol provided for InDesignServer communication is not (yet) supported.

Solution: use LOCAL for InDesignServer instances running on the priint application server machine (Corba will be used for communication in that case) or REMOTE for instances on remote machines (SOAP over HTTP will be used then).

DOCUMENT_NOT_OPENED (1227)

Description: the document is not opened. Either the InDesignServer istance responsible for requests regarding this document lost the reference to that document or your server is configured to disallow *implicitOpen* and you missed to call *documentOpen* before operating on the document.

Solution: check the server configuration or your client application.

BINARYTRANSPORT_ERROR (1228)

Description: something went wrong while trying to pass binary data from InDesignServer to the client application.

Solution: most likely this has something to do with unsufficient filesystem privileges. To avoid permission problems InDesignServer and priint:comet server should run under the same user account.

INVALID_PARAMETER_FORMAT (1229)

Description: a value of a certain type and range was expected but you provided something different.

Solution: check the documentation for valid arguments. If the documentation appears to be wrong in that point we appreciate your feedback!

NOTYETIMPLEMENTED (1230)

Description: the API is available but the method is not yet implemented.

Solution: this should not happen and if this is related to a documented API method we appreciate your feedback.

CLASSINSTANTIATION_ERROR (1231)

Description: dynamic delegation to a class failed, because this class is not available or failed to load.

Solution: never thrown at the moment

PATH_EXCEPTION (1232)

Description: a path required for an operation could not be setup. This could be a session temporary directory, the base directory for dataset default file storage etc.

Solution: check configuration and filesystem privileges.

DOCUMENT_EXISTS (1233)

Description: you try to overwrite a document that already exists and your server policies require to delete that document first.

Solution: check configuration or change the client application to reflect the server policies.

DOCUMENT_SAVE_ERROR (1234)

Description: a document could not be written to filesystem.

Solution: most likely a filesystem privileges issue.

DOCUMENT_IN_USE (1235)

Description: a document cannot be deleted or overwritten because it is currently opened or checked out.

Solution: check if the document is opened in other sessions (see status in the document table)

DOCUMENT_CLOSE_TIMEOUT (1236)

Description: a document could not be closed because the responsible InDesignServer instance was not available.

Solution: check if the instance is running and maybe blocked by a long running job.

NOT_A_TMPFILE (1237)

Description: where a temporary document was expected the document provided was something else.

Solution: you cannot apply this operation to that document

SERVER_STATE_EXCEPTION (1238)

Description: something went seriously wrong which could inidcate that the server is in an illegal state.

Solution: refer to logfiles, most probably you will also have to check configuration (FS permissions etc.)

NO_DATABASE_CONNECTION (1239)

Description: an operation requires a database connection but there is none available.

Solution: check the connection pool and jdbc resource configuration, check the persistence.props file.

COMET_SOAP_CLIENT_EXCEPTION (1240)

Description: a SOAP connection to a comet data provider is required but not available.

Solution: at the moment there is nothing you can do.

DOCUMENT_NOT_REGISTERED (1241)

Description: a document ID could not be resolved. The document has neither been registered nor a publication system could provide a path for that document ID.

Solution: if you think the document ID is correct check the document table in the admin database or your publication system if the document exists. Check if the session is connected to the correct dataset.

DOCUMENT_NOT_UNIQUE (1242)

Description: a document ID is not unique.

Solution: until now this exception never occured. Something must be seriously broken in your installation.

TARGET_INVOCATION_EXCEPTION (1243)

Description: this is a wrapper for exceptions thrown by third party modules and components or modules and components not in the standard distribution of comet server.

Solution: refer to the documentation of the target system for further description and analysis.

REPOSITORY_NOT_FOUND (1244)

Description: a repository could not be found.

Solution: does the requested repository exist?

DOCUMENT_LOCKED (1246)

Description: a document could not be used because it is currently locked.

Solution: most probably the document is checked out by a client application.

PLEASE_CHECK_CONFIGURATION (4711)

Description: an operation required a resource that is not available in your configuration.

Solution: please check your configuration.

E.2 priint:comet server plugins Exceptions

Some of these exceptions are handled by priint:comet server so they will never be passed to the client application, though you might stumble accross them when reading server logs. The following list should help you to better track down error reasons.

COMET_CSCRIPT_ERROR (538624)

Description: a cscript returned an error.

Solution: see the PlugIns logfile, there you should find a hint where the error occured. If its caused by malformed syntax fix the script, if certain preconditions were not met when executing the script the error reason could as well be other misconfiguration or an error in the client application.

COMET_NO_LICENSE (538625)

Description: no license could be found for the PlugIns / Version you installed.

Solution: you should see a license order in the console output of the InDesignServer instance or in the PlugIns logfile. Send this order to license@priint.net, install the license file and restart InDesignServer.

NOT_YET_IMPLEMENTED (538626)

Description: the method requested is not yet implemented in the PlugIn Version you are using.

Solution: if this is a documented API method and you have installed the latest priint:comet server and PlugIn releases this should not happen.

INITIALIZATION_ERROR (538627)

Description: an error occured while initializing an InDesignServer instance

Solution: check the "CometEnvironment" folder. This should be a valid priint:comet XML folder. Could be a problem with any of the XML files, broken PDF and printer profiles, filesystem privileges.

DOCUMENT_NOT_FOUND (538628)

Description: a document could not be found

Solution: check, if the documentID / path resolution is correct. Is the path provided by priint:comet server correct? Is this path reachable from the machine / account InDesignServer is running on?

ILLEGAL_DOCUMENT_STATE (538629)

Description: the operation could not be applied to the document because it is in an unexpected state.

Solution: please refer to the PlugIns / API log what happened to that document before. Most probably you will also have to restert the InDesignServer instance the document is opened on.

DOCUMENT_OUTDATED (538630)

Description: a operation could not be applied to the document because the target (spread, page, cometgroup, element etc.) is not available anymore or does not allow this operation.

Solution: check if your client application reflects all changes made to a document

DOCUMENT_VANISHED (538631)

Description: a document expected to be opened on a certain instance is not opened anymore.

Solution: most probably this occured after several server restarts and the document statuses are out of sync. If possible you should consider to restart all services. Also check the entries in the comet_admin.adm_document table.

MALFORMED_XML (538632)

Description: a XML provided by the user is malformed.

Solution: this can be a *setNotes* or *build* operation. Check if the XML generated in your client application is correct.

INVALID_XML (538633)

Description: the XML is well formed but invalid.

Solution: see MALFORMED_XML.

PAGE_NOT_FOUND (538634)

Description: a target page could not be found.

Solution: is the page index provided correct (indexes are *zero* based)? Is the document structure information in your client application up to date?

INVALID_BOUNDS (538635)

Description: the bounds provided for a build operation (such as placeTemplate...) are incorrect.

Solution: check bounds.

SPREAD_NOT_FOUND (538636)

Description: a target spread could not be found.

Solution: is the spread index provided correct (indexes are *zero* based)? Is the document structure information in your client application up to date?

GROUP_NOT_FOUND (538637)

Description: a target group could not be found.

Solution: is the cometgroup ID provided correct? Is the document structure information in your client application up to date?

ELEMENT_NOT_FOUND (538638)

Description: a target element could not be found.

Solution: is the element ID provided correct? Is the document structure information in your client application up to date?

PLACEHOLDER_NOT_FOUND (538639)

Description: a target placeholder could not be found.

Solution: is the placeholder ID provided correct? Is the record ID provided correct? Is the document structure information in your client application up to date? Do the PlugIns use the correct database connection?

UNKNOWN_PDF_PROFILE (538640)

Description: a pdf profile name provided by the user could not be found in the InDesign prefs.

Solution: if it is a profile managed in the pdfprofiles.xml file (CometEnvironment): did InDesignServer initialize properly? You should see a list of loaded PDF profiles in the API log right after the *setEnvironment* call.

INVALID_TEMPLATE_ID (538641)

Description: the template ID provided is not valid.

Solution: check if the PlugIn is connected to the correct database.

INVALID_SETUP_PARAMS (538642)

Description: the format or values provided for document setup are invalid.

Solution: check log files and client application.

CONNECTION_REQUIRED (538643)

Description: the operation requires a database connection but none is available.

Solution: this should not happen, check if you use the latest priint:comet server and PlugIn releases and if your connection settings are correct.

UNRECOVERABLE_ERROR (538644)

Description: this error indicates serious problems with your InDesignServer installation. It will be thrown if no memory can be allocated or very basic InDesign operations (such as getting the spread list of a document) fail.

Solution: restart InDesign, if the problem still occurs check the installation.

BAD_POSITION (538645)

Description: coordinates provided are invalid.

Solution: refer to log file (api.log) for further hints.

INVALID_GRID_ID (538646)

Description: the grid ID provided is not valid.

Solution: check connection configuration, check client application.

GENERIC_BUILD_ERROR (538647)

Description: an error occured while building products.

Solution: refer to the PlugIns logfile for further hints. Could be broken configuration, missing images, filesystem issues etc.

ILLEGAL_OPTIONS (538648)

Description: the options provided for a command are invalid.

Solution: see the developer documentation, section "Options" for more information

UNKNOWN_SERVICE (538654)

Description: the service name provided for an ODBC connection is unknown.

Solution: check configuration, in particular the connections.xml file in your CometEnvironment folder. CheckDSN configuration.

AUTHORIZATION_ERROR (538655)

Description: the credentials provided for a database connection are invalid.

Solution: check configuration, in particular the connections.xml file in your CometEnvironment folder. CheckDSN configuration.

UNKNOWN_POOL (538656)

Description: a datapool could not be found.

Solution: check configuration, in particular the connections.xml file in your CometEnvironment folder.

CONNECTION_FAILED (538657)

Description: connecting to a database failed for any other reason but the errors described above.

Solution: check configuration, in particular the connections.xml file in your CometEnvironment folder. CheckDSN configuration.

SERVER_WAS_DOWN (538658)

Description: indicates, that an InDesignServer instance was shutdown since the time it was registered by priint:comet server.

Solution: this exception is handled by comet server, the InDesignServer instance will be reregistered.

UNKNOWN_EXPORT_FORMAT (538659)

Description: the export format provided is not valid.

Solution: at the time only "plain" and "tagged" and "raw" are supported.

UNKNOWN_MASTERPAGE (538660)

Description: the masterpage provided is invalid

Solution: provide a valid masterpage. Avoid special characters in masterpage names, in certain environments this can cause trouble.

ILLEGAL_ARGUMENT (538674)

Description: the format of a parameter does not match what was expected.

Solution: check if you use the latest releases of both priint:comet server and PlugIns.

E.3 InDesign ErrorCodes

Sometimes InDesign ErrorCodes are propagated to the client application. A full list of InDesign error codes can be found in the Adobe InDesign Products SDK documentation (docs/references/errorcodes.htm).

These are the most important:

FAILED (1)

Description: this appears to occur rather often

Solution: check the comet log file for a more detailed error description

kOpenDocFailedError (3588)

Description: document could not be opened - probably required Plug-Ins are missing or your InDesign version is too old

Solution: check file, InDesign and Plug-Ins (also other 3d party Plug-Ins) versions

kSaveDocFailedError (3590)

Description: saving a document failed. Most probably the document was converted from an older InDesign version or you don't have write permissions on this file.

Solution: check file, InDesign and Plug-Ins (also other 3d party Plug-Ins) versions

kTooManyOpenFilesError (268)

Description: too many opened InDesign documents. Note: could also be opened templates.

Solution: change your client application to keep the numbers of opened files small. If this is caused by too many opened (Comet-) templates, please contact support@priint.com