



OLicense-Server 4.8

(V4.8.0)

Manual

Optimum GmbH
Hirschstr. 12-14
D-76133 Karlsruhe

E-Mail: office@optimum-gmbh.de
Internet: <http://www.optimum-gmbh.de>

1 Table of contents

| | | |
|--------|--|----|
| 1 | Table of contents | 3 |
| 2 | List of figures | 5 |
| 3 | Welcome | 6 |
| 4 | General information | 7 |
| 4.1 | About this manual..... | 7 |
| 4.1.1 | Conventions | 7 |
| 4.1.2 | Abbreviations and definitions | 8 |
| 4.1.3 | Pictograms | 8 |
| 4.2 | User roles referenced in this manual..... | 9 |
| 5 | Interaction of Olicense-Suite roles..... | 10 |
| 6 | Language support..... | 11 |
| 6.1 | Language selection | 11 |
| 7 | Installing the OLicense-Server | 12 |
| 7.1 | System requirements | 12 |
| 7.2 | Starting the installation process..... | 12 |
| 8 | Uninstalling the Olicense-Server | 13 |
| 8.1 | Starting the uninstallation process..... | 13 |
| 9 | Overview | 13 |
| 10 | Starting and Stopping Olicense-Server..... | 14 |
| 10.1 | General information..... | 14 |
| 10.2 | Default startup | 15 |
| 10.3 | Starting with an arbitrary TCP/IP port number..... | 16 |
| 10.4 | Starting with arbitrary database path settings | 17 |
| 10.5 | All start options and INI file..... | 17 |
| 10.6 | All Controller-Startparameter..... | 21 |
| 11 | Administrating OLicense-Server | 23 |
| 11.1 | Use case "License Request" | 24 |
| 11.2 | Use case "License Import" | 24 |
| 11.3 | Use case "License Report"..... | 25 |
| 11.4 | Use case "Server Administration"..... | 27 |
| 11.5 | Use case "License Check In/Out"..... | 30 |
| 11.5.1 | Checking out licenses | 30 |
| 11.5.2 | Checking in licenses | 31 |
| 11.6 | Use case „License Administration“ | 31 |
| 11.6.1 | Unload licenses..... | 32 |

| | | |
|--------|--|----|
| 11.6.2 | Changing the OLicense-Server's hardware..... | 33 |
| 11.6.3 | Enable/disable licenses..... | 34 |
| 11.6.4 | Activate/deactivate licensee feedback (supply additional information) | 34 |
| 11.6.5 | Check out settings..... | 34 |
| 11.6.6 | Group-Pool..... | 34 |
| 11.7 | Sub use case „Change check out privileges“ | 34 |
| 11.8 | Sub use case „Admin Group Floating Licenses“ | 36 |
| 11.9 | Use case „Floating License Reservation“ | 39 |
| 11.9.1 | Examples for reservation..... | 39 |
| 11.10 | Use case "Group Floating Blacklist" | 40 |
| 12 | High availability server system..... | 41 |
| 12.1 | Server system configurations | 42 |
| 12.1.1 | Optimal..... | 42 |
| 12.1.2 | Minimal..... | 42 |
| 12.1.3 | Critical | 42 |
| 12.1.4 | Mixed Mode..... | 43 |
| 12.2 | Starting/Stopping of servers in a server system | 43 |
| 12.3 | Starting/Stopping of proxies in a server system | 43 |
| 12.4 | Second High-Availability Variant | 45 |
| 13 | Backup and Restore | 47 |
| 13.1 | Administration of database files..... | 47 |
| 13.2 | Backup database files | 47 |
| 13.3 | Restoring database files..... | 47 |
| 13.4 | Reinstallation of OLicense-Server | 48 |
| 13.5 | Backup in batch mode..... | 49 |
| 14 | (U3)-USB-Memory-Stick | 50 |
| 14.1 | Specifics for Linux | 50 |
| 15 | Serverstatistics..... | 52 |
| 16 | olixtool – the command line tool..... | 52 |
| 17 | Appendix: Messages..... | 55 |
| 17.1 | Transaction errors | 55 |
| 17.1.1 | Communication failure | 55 |
| 17.1.2 | Protocol error | 56 |
| 17.1.3 | OLicense-Server errors | 57 |
| 17.2 | Transaction ok..... | 58 |
| 17.3 | Error (exception) caused by erroneous programming with OLicense API .. | 58 |
| 18 | Further information..... | 59 |

2 List of figures

| | |
|---|-----------|
| Figure 1: OLicense-Server Controller | 15 |
| Figure 2: Olicense-Server Controller; Local start | 16 |
| Figure 3: OLicense-Server administration screen | 23 |
| Figure 4: Committing successful license uploading | 25 |
| Figure 5: Error message caused by an erroneous license uploading | 25 |
| Figure 6: License report example | 26 |
| Figure 7: OLicense-Server configuration window | 28 |
| Figure 8: License administration | 32 |
| Figure 9: OLicense-Server's response after license removal (except for floating and pay-per-use licenses) | 33 |
| Figure 10: Removal code for floating and pay-per-use licenses..... | 33 |
| Figure 11: Admin Group Floating License | 37 |
| Figure 12: Floating license reservation..... | 39 |

3 Welcome

With this software package you should have purchased OLicense-Server which is a tool for license management.

There is no need to add a second instance of OLicense-Server to an existing framework if there is already an OLicense-Server running.

OLicense-Server is one component of OLicense-Suite, a modern tool for software license management and user rights deployment in a network environment.

Further information and the latest server release for download is provided at <http://olicense.com>.

We use the term „**server**“ analogous to other popular web-servers like Apache-HTTP-server because OLicense-Server is a stand-alone web-server. On a windows environment the server is just another kind of service, in a UNIX-like environment it's a daemon. The server can be installed either locally on a notebook or on a server-host in the internet (notice the ambiguity of the term “server”).

The OLicense-Server can be installed and administrated easily. In the best case scenario the software-manufacturer has used all features of the OLicense-Suite to minimize your administration costs.

4 General information

4.1 *About this manual*

4.1.1 Conventions

In this manual the following typographic conventions are used:

| Object | Appearance |
|---|---|
| Names for menus, tabs, buttons, dialogs, fields, paths, files and so on | Font: Times New Roman (C:\Programs\Optimum\Olicense) |
| Commands, code samples | Font : Courier (cd c :\\programme\...) |
| Compulsory values to be defined by the user | <in angle brackets> |
| Optional fixed values | [in squared brackets] |
| Optional values to be defined by the user | [<in angle and squared brackets>] |

Input for command shells is described for Windows operating systems. In Linux the string `./` has to be prepended and the extension for executables has to be converted from `.exe` to `.lin`.

Example:

on Windows:

```
OLicenseServerCtrl.exe
```

on Linux:

```
./OLicenseServerCtrl.lin
```

4.1.2 Abbreviations and definitions

| | |
|-------------|---|
| MajVersMin | Minimum major version of a software product. |
| MajVersMax | Maximum major version of a software product. |
| POP | Post office protocol; a protocol used to receive e-mails from a mail server. |
| SMTP | Simple mail transfer protocol; a protocol used to send e-mails to a mail server. |
| Server | Service or daemon. A software program, that runs in the background and is started automatically during the system boot process. |
| Server host | The network computer on which the server programs are running. |
| SW | Software |
| XML | Extended Markup Language; a data format to represent structured data as a string. |

4.1.3 Pictograms



Hint



Attention, important notice

4.2 *User roles referenced in this manual*

| Name(s) | Description |
|------------------------------|---|
| Software-manufacturer | Person or organization, who manufactures software products. |
| Distributor, Software-vendor | Person or organization, who ships the software product to it's customer. This may be the software manufacturer himself or a distribution organization. |
| Licenser | Person or organization, who generates and ships licenses for licensed software products. This may be a person or group within the software manufacturer or the distributor. |
| Licensee | Person or organization utilizing the licensed software product i.e. generally you |

5 Interaction of Olicense-Suite roles

This scenario matches, if your software-manufacturer has utilized only the minimal possible functionality of Olicense-Suite. Your manufacturer could be able to extend the system to a fully automatic machine which implies that you only have to insert your customer data in a dialog box of the software.

| Licenser | Licensee |
|---|--|
| | Customer orders software product |
| Ships the software product and the Olicense-Server to the customer. | |
| | Installs the software product and the OLicense-Server and returns the licensee and server information required for license generation to the licenser. |
| Creates a license key and ships it to the licensee. | |
| | Uploads the license file into his Olicense-Server and utilizes the software product. The OLicense-Server checks the availability of a license according to the information in the uploaded license file. |

6 Language support

| | Graphical User Interface | Manual |
|---------------------------------------|-------------------------------|-------------------|
| Olicense-Server and Controller | German English Japanese | German English |

6.1 *Language selection*

OLicense-Server and its graphical Controller frontend are multilingual. The used language is specified by the operating system of the computer, i.e. on a German operating system installation, the programs will use German language, on an English operating system installation, it will use English language. The default language, i.e. if all dictionaries are missing, is English.

You can force the Server and Controller to start on a different language if you call the tool with the following commandline-parameter (for the Server you can also append the INI-file with): `-lang <language id>`. Currently the language ids *de* for German, *en* for English and *ja* for Japanese are supported. See 10.5 All start options and INI file.



Other languages can be supported. Please contact us, if you need help for other languages or if you want to create a dictionary for other languages yourself.

7 Installing the OLicense-Server

7.1 System requirements

For a proper operation of OLicense-Suite components, the following system resources are recommended:

| | Olicense-Server |
|---|--|
| Processor | Pentium min. 600 MHz |
| Operating system tested with: | <i>(see the actual list on olicense.com)</i> |
| Hard disk: | Approx. 15MB |
| RAM: | Approx. 6 MB (real free) |
| Screen resolution: | Arbitrary |
| Network protocol | TCP/IP |
| Internet Browser tested with: | Mozilla Firefox >14 |

7.2 Starting the installation process

Required privileges:

local administrator privileges on Windows, root privileges on UNIX if you want to install the server as service/daemon.

Call the installer application. On Windows:

`OLicenseServer-4.8...-Setup.exe`. On Linux:

`OLicenseServer-4.8...-Setup.lin`.

Follow the instructions.



OLicense-Server has to be installed on a local drive of a network ready computer.

8 Uninstalling the Olicense-Server

8.1 *Starting the uninstallation process*

Uninstallation of the software can be accomplished by:

 Start -> Settings -> System -> Software (Windows)

respective

SuSE/Gnome-Menu -> OLicense-Server -> Uninstall (Linux).

9 Overview

OLicense-Server is a special type of web server. It provides a web service that supervises the utilization of licensed software in an intra/internet environment. It uses state of the art encryption techniques to stem misuse. Installation and administration can be accomplished easily without requiring extensive system resources. It uses a widespread network protocol (TCP/IP) in combination with XML/SOAP as transmission protocol for selective conveyance bypassing firewalls.



In a Windows-XP and higher environment the **firewall** is blocked by default. To enable external requests to the server, you have to open the firewall for the server.

10 Starting and Stopping OLicense-Server

10.1 General information

If there is already one OLicense-Server running in your network, you don't have to install a second one. One instance is sufficient.



To install OLicense-Server as service/daemon you need Administrator-privileges !



OLicense-Server and the corresponding database files have to be installed on a local disc.



OLicense-Server is, if installed as service/daemon, started automatically by the operating system during the system boot process.

You can administrate OLicense-Server with the OLicense-Server-Control frontend either by invoking the corresponding entry in your start menu or by clicking the desktop icon if installed:

(Start -> Programs or SuSE.Menu)-> OLicense-Server -> OLicense-Server-Control)

Once installed as service/daemon, the server can be controlled also using the Windows service dialog. This dialog can be opened using

Start -> Settings -> System -> Administration -> Services

- On Linux you can control OLicense-Server with /etc/init.d in a similar fashion as the other servers in your system. Further information is provided by the operating system manual.



If your OLicense-Server supervises licenses distributed on different clients in a network environment, it is recommended that your host computer runs permanently. Special attention has to be paid to following items:

- Ensure that OLicense-Server is already started when licensed software is called.
- Your host computer doesn't require extensive system resources to run OLicense-Server but high stability and availability is strongly recommended. If using critical applications you can establish an emergency server system consisting of several redundant OLicense-Servers. Further information can be found on chapter 12:

Highest availability servers on page 41.

Alternatively you can install a local copy of OLicense-Server and your licensed software to use it for instance on a mobile computer.



If possible, do not run OLicense-Server on a host computer that operates another web-server which usually uses HTTP port 80 by default. If this cannot be avoided you can start OLicense-Server on another port, e.g.: 8080. To start OLicense-Server on a different port number see chapter 10.3 *Starting with an arbitrary TCP/IP port.*

10.2 **Default startup**

After installation you can start OLicense-Server-Control frontend. If you wish to install the Server as service/daemon you must call the Controller with Administrator/root rights. On Linux an additional button is visible to obtain root rights if necessary tools like kdesu are installed.

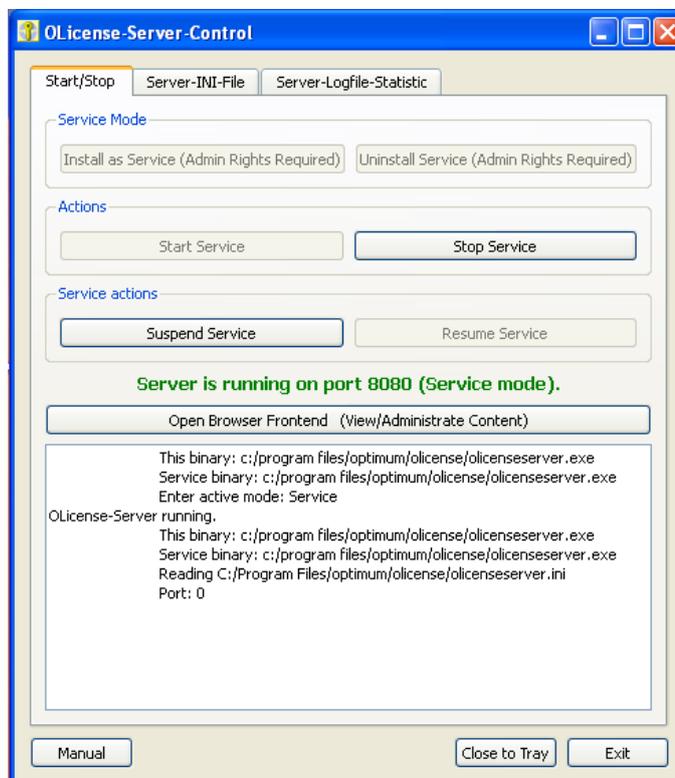


Figure 1: OLicense-Server Controller

In the Start/Stop tab you can de/install the server as service/daemon, you can start/stop an installed service, or you can start/stop the server serviceless.

If you click the button: Open Browser Frontend, the Controller tries to start your local installed Internet-Browser and opens the homepage of OLicense-Server.

If you installed the server as service you may close the Controller, it is not needed for the operation. If the server is started serviceless, the Controller must be active but you may 'park' it in the system tray of your operating system by clicking: Close To Tray.

You may operate more than one server on one machine, but only one can run as service and only one can serve Floating licenses. If you try to start a second server you will get an error message that you can start only if you deactivate Floating licenses.

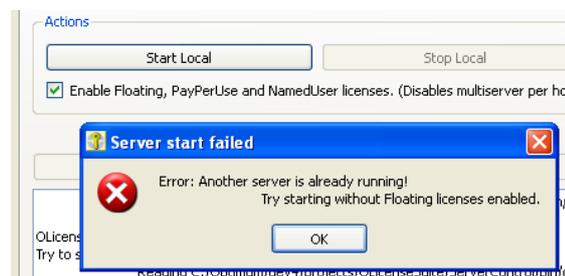


Figure 2: OLicense-Server Controller; Local start

In this case uncheck the checkbox: Enable Floating, ..., and start again.

10.3 Starting with an arbitrary TCP/IP port number

In the Controller tab: Server-INI-File, you may define miscellaneous start parameters.

During server start the Controller will inform you if the current port is in use. In this case you must change the parameter: `-port` in the INI-file. Set another port number and click: Save. Switch back to the Start/Stop tab and try to start the server again.

It can be possible that a port is reported as 'in use' but no application is using it. This can happen if e.g. a program crashed and the operating system needs some time to deblock. In this case the start dialog permits you to start the server on that port by force.

10.4 *Starting with arbitrary database path settings*

To save the database files of OLicense-Server (OLicenseServerDB.odb and OLicenseServerDB.odx) in a local directory different from the installation directory, add parameter `-dbFile <path>` to your command line.

Proceed as described in chapter 10.3 (Starting with an arbitrary TCP/IP port, page 16) to access the command line and append the desired database path to your program call.



You may only use forward (/) slashes in your database path. Filename extensions have to be omitted as well.

10.5 *All start options and INI file*

All start options are listed below:

| Name | Meaning |
|--|---|
| <code>-timeout <sec></code> | Maximum period of time the server is allowed to wait for client data. |
| <code>-port <number></code> | Port where the server is listening, see Chapter 10.3 |
| <code>-dbFile <filename></code> | Database file for the server, see Chapter 10.4 |
| <code>-logFile <filename></code> | A file to store special debug information. This file can give details if a (composit)server does not start orderly. Please attach this file if you send a support request to Optimum. |
| <code>-proxyOnly</code> | If you are going to use OLicense-Server in an emergency server system and direct client requests have to be suppressed by your server. See also <code>-fallback...</code> |
| <code>-servers <s1>[,<s2>,<sn>]</code> | If the server is intended to be used as a proxy-server in an emergency server system. See also <code>-fallback...</code> |
| <code>-denyReportCols</code> | Suppresses the display of the specified columns |

| Name | Meaning |
|---------------------------|--|
| <n1>[,<n2>,<nn>] | in your license report. |
| -lang <country code> | <p>Starts the server in the given language. Default is: en. For other codes the corresponding dictionaries must be present:</p> <p><code>olicenseserver_<code>.qm</code>, e.g.: <code>olicenseserver_de.qm</code> for German.</p> |
| -allowOccupancyInfoGlobal | <p>By default the server information: „Who has currently locked which Floating-License“, is blocked for privacy protection. This information can be released separately for every license via the License Administration page. Is a global release for all licenses desired then use this parameter.</p> |
| -respondHttpErrors | <p>HTTP errors will be send without the triggered question in the error message per default. Setting this parameter will embed the question.</p> |
| -deviceWait <sec> | <p>If a given database file is located on a (e.g. U3) device which is not available during server startup, the server will test every 3 seconds up to the given seconds if the device is now available. (On Windows the server will wait 21 seconds per default even if this parameter is not given)</p> |
| -noAutoBoot | <p>Prohibits under Linux that autoboot-scripts are automatically created in /etc/init.d during server-startup. This is of interest if own boot-scripts must be used.</p> |

| | |
|--|--|
| <p>-initialPw <MD5 user:pw></p> | <p>Some server web pages can be protected by a password to avoid unmeant access in the licensee's network. On delivery there is no password set. Via this parameter an initial password can be set, that will be active as long as no password is set via the server's admin page. The parameter's value is the HEX string of the MD5 Hashsum of the string: <user>:<password>, e.g.: Hugo:MyPW (see also -makeInitialPw)</p> |
| <p>-makeInitialPw <user>:<password></p> | <p>Creates the MD5 Hash needed by -initialPw Output takes place on STDOUT</p> |
| <p>-allowedFrontEndPeers <TCP-Addr>[;<TCP-Addr2>;...]</p> | <p>Defines from which network addresses it is allowed to access the server's web pages. The addresses must be separated by ';'. Wildcards (*) are allowed, e.g. -allowedFontEndPeers 127.0.0.1;192.168.*.*</p> |
| <p>-runAs <login name uid:gid></p> | <p>(Linux only!) If the server starts with root rights, it will switch automatically to the given user account.</p> |
| <p>-u3ScanOnly <dev 1>[;<dev2>;...]</p> | <p>Usually the server scans cyclic all connected USB/U3-disks. If a list of U3-IDs is given with this parameter, the server scans in the following only those drives on which an ID is found. This is helpful if single USB disks would lose performance through a scan access.</p> |
| <p>-fallbackMyself <tcp-addr>[:<port>] -fallbackServer1 <tcp-addr>[:<port>] -fallbackServer2 <tcp-addr>[:<port>]</p> | <p>All three parameters must be given to create an emergency server system in a second variant. See chapter 12.4. The tcp-addr must be in numerical form.</p> |

| | |
|-------------------------|---|
| -initialLogPath <path> | As long as no path is written for the protocol file to the database, the server writes the protocol file in its installation directory. This parameter changes this behaviour and forces the protocol file to the given path. |
| -overrideLogPath <path> | This parameter forces the protocol file to the given path. The path from the database is ignored. |

You can get additional information by inspecting the sample INI-file included in your installation.

10.6 All Controller-Startparameter

The graphical controller dialog can also be started with parameters.

All start options are listed below:

| Name | Beschreibung |
|-------------------------------------|--|
| -start [-enableFloating true false] | <p>After the controller dialog is started, he tries to start the configured server automatically. On success the controller will be minimized to system tray. On failure the dialog will be left open and will show an error message.</p> <p>If a controller dialog is already open during execution of this command, the command will be forwarded to this controller.</p> <p>The additional parameter <i>-enableFloating</i> defines whether the server shall be started with or without Floating licenses enabled. This has an effect only if a second server is already running on the same operating system, see Figure 2: Olicense-Server Controller; Local start.</p> |
| -stop | <p>Stops the server which was started via the controller. An open controller must exist during execution of this command.</p> |
| -import <filename> | <p>Passes filename to the controller. The file can then be imported in the server with a click.</p> |
| -raise <leer> | <p>Starts the controller or moves an existing controller to the foreground, if the operating system allows it.</p> |

| | |
|----------------------------------|--|
| <p>-lang <language id></p> | <p>Forces the Controller to start with the given language. Precondition: A dictionary of the language must be present.</p> <p>Possible languages:</p> <p>en: English</p> <p>de : German</p> <p>ja : Japanese</p> |
|----------------------------------|--|

11 Administrating OLicense-Server

The fastest way to the administration window of OLicense-Server is to click the button: Open Browser Frontend, on the Controller.

Only if you do not have a registered standard browser in your operating system, you need to start your browser by hand. Set the following URL:

```
http://localhost
```

If a different port number is preferred or if you want to open the window from an arbitrary computer that is not the host computer - call your browser and set the URL:

```
http://<name>[:<Port>]
```

where <name> is the name or the TCP/IP address of your host computer and <port> is the port number being used by OLicense-Server. If port 80 is used the -port option may be omitted.

E.g.: `http://www.foo.bar:8080`

This is the administration window of OLicense-Server:



Welcome to the
OLicense-Server

| There are seven use cases to work with me: | | |
|--|--|---|
| License Request | Grants or refuses a license request from a software program. The request protocol is SOAP (Simple Object Access Protocol) with Rijndael ciphered data. | (A licensed software program with the OLicense-Client-API compiled in will communicate this way.) |
| License Import | Imports a license key file got from your licensor. The license file must be in XML<IDOCYTYPE OLicense> format. The keys are needed to validate a license request. | <input type="text"/> <input type="button" value="Browse..."/> <input type="button" value="Import Selected XML"/> Use this form to select and send the license file. |
| License Report | Shows the imported licenses. | <input type="button" value="View Licenses"/> |
| License Check In/Out | Calls a password protected form to check out licenses from a remote server, or to check in the checked out licenses. | <input type="text"/> URL of the remote server <input type="button" value="Check Out Licenses..."/> <input type="button" value="Check In Licenses..."/> |
| Server Administration | Calls a password protected form to manage some server attributes. | <input type="button" value="Administrate Server..."/> |
| License Administration | Calls a password protected form to configure and delete licenses. | <input type="button" value="Administrate Licenses..."/> |
| Floating License Reservation | Calls a password protected form to reserve floating licenses to hosts or users. | <input type="button" value="Reserve Floating Licenses..."/> |

Figure 3: OLicense-Server administration screen



If you cannot establish a connection with your OLicense-Server, this may be caused by the proxy or firewall settings of your browser. For more

information see chapter **Fehler! Verweisquelle konnte nicht gefunden werden. Fehler! Verweisquelle konnte nicht gefunden werden.** on page **Fehler! Textmarke nicht definiert..**

The following chapters deal with use cases for OLicense-Server-administration.

The use cases License Check In/Out, Server Administration, License Administration, Floating License Reservation and Group Floating Blacklist are password protected.



The corresponding fields have to be left empty unless you specify administrator's name and password in the authorization dialog.

Each use case offers a chance to return to the main window of OLicense-Server administration by clicking the link Back to Homepage or the Back button of your browser.

11.1 **Use case “License Request”**

This use case is automatically applied when a licensed software requests or releases a license. There is no possibility or need for manual interaction.

11.2 **Use case “License Import”**

Since version 3.2.8 the server registers olixml file extensions on a window platform as import-files, meaning that you can import license-files into OLicense-Server just by double clicking them in the windows-explorer. The same applies for license-import from an e-mail application.

Another possibility is, that the manufacturer has implemented a separate import-dialog box in his software-product. In this case, manual importation is not required.

For all other cases importation has to be done in the mainwindow of the server.

At first save the license file, that you have received from the licenser, on your computer. Then press the button `File open` in the OLicense-Server administration window and select the license file in the appearing file select dialog.



Ensure to set File type to All files (*.*) otherwise you will not see the XML license files.

Confirm your selection by pressing the Ok button. The path of the selected file is transferred to the corresponding field in the administration window. To upload the licenses to the OLicense-Server press button Send.



If a license is changed or extended, the old license entry will be replaced during the upload process.



Expired licenses are automatically removed from OLicense-Server's database.

Successful uploading of a license file will be committed by the OLicense-Server:

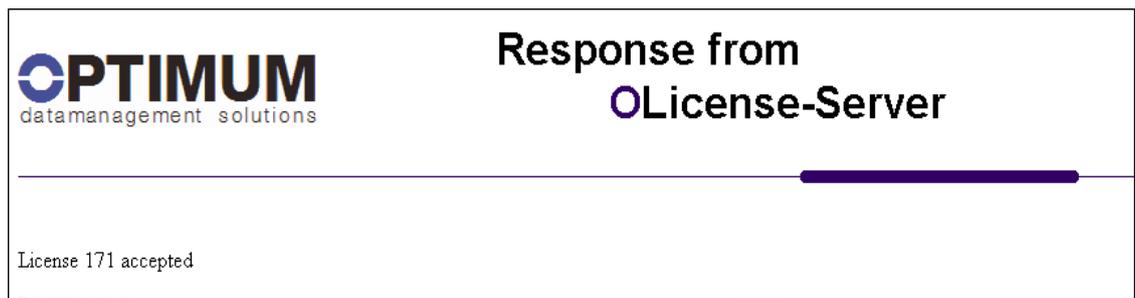


Figure 4: Committing successful license uploading

If one or more licenses in the license-file could not be uploaded the following error message occurs:



Figure 5: Error message caused by an erroneous license uploading

11.3 Use case "License Report"

Clicking the button View Licenses leads to a listing of all licenses stored in the OLicense-Server's database:



Response from OLicense-Server

| application | version | module | options | license type | floating (in use) | users | hosts | expiration | mod |
|----------------|---------|----------------------|---------|--------------|-------------------|-------|--|------------|--------------------|
| Test | 0-0 | print_id, print_user | | Demo | | | | Infinite | Test-E84D226 |
| OLicense-Suite | 4-4 | core | MU | NodeLocked | | | 001438039 aTRZohPig bWMmk6QzEFtPoCXkEK | Infinite | Optimum 626BE06 |

Figure 6: License report example

Description of the license report columns:

Attention! Be aware that you can hide columns by setting the corresponding start option.

| Column | Description |
|--------------------------|---|
| Application | Name of the licensed software |
| Version | Version range of the licensed software for which the license is valid. If this information is missing, the license is valid for all versions. |
| Module | Names of the licensed features/modules of the licensed software. If the license is valid for all modules this field is empty. |
| Options | Names of the licensed options of the licensed software. Unlike Modules options do not require extra licenses |
| License Type | License type, e.g. Floating. |
| Floating (in use) | Number of concurrent users available for a floating license. The value in brackets shows the number of currently allocated licenses. If two values are shown the first entry represents the number of checked out licenses. For other license models this field is empty. |
| Users | Licensed users of named-user licenses. For other license types this field is empty. |

| Column | Description |
|---------------------------------|---|
| Hosts | Computer ID's for node-locked and pay-per-use licenses. For other license models this field is empty. |
| Expiration date | The last day the license is valid. Values of oo or 7999-12-31 represent an unlimited validity. |
| License | Identifier of the license data record. |
| Licenser | Licenser's name. |
| Licensee | Licensee's name. |
| Key | Identifier of the licensed software. |
| Upload date | Date at which the license was uploaded. |
| Group size/ Lifetime | For Group Floating licenses. Shows the pool size and the lifetime of the pool entries. |

11.4 Use case “Server Administration“

Pressing button Administrate Server opens a dialog where you can modify your OLicense-Server configuration:

| OLicense-Server Attributes | | |
|--|--|--|
| <input type="checkbox"/> User & Password | Sets/Changes the username and password which protects most forms. | User <input type="text"/> Password <input type="password"/> Confirm <input type="password"/> |
| <input type="checkbox"/> Logfile | Enables/Disables the logfile. Sets the path where the server will store the logfiles. A logfile contains the session protocol (e.g. who requested a license and when). The filename is: OL<YYYYMM>.log. | Logfile enabled Yes <input checked="" type="radio"/> No <input type="radio"/> Path C:/Program Files/optimum/olicense/ |
| <input type="checkbox"/> Alert Email | Sets the alert email address and the mail server. The license server will send an email to that address if e.g. a license expires etc. | eMail address <input type="text"/> Smtp server <input type="text"/> 25 Warn #weeks before a license expires 2 |
| <input type="checkbox"/> PayPerUse Filepath | Sets the path where the server will store the protocol files for a PayPerUse license. The filename is: OL_<Licenser-Ident>_<YYYYMM>.ppu. | C:/Program Files/optimum/olicense/ |
| <input type="checkbox"/> Check In/Out Server | Sets the URL of the preferred server for license check in and out Format: <protocol>://<server>:<port> e.g. http://localhost:80 | <input type="text"/> |
| Preferred Server ID | The preferred server ID. Used by Floating, PayPerUse and optional by NamedUser licenses. If it is not visible on the right side, one of the alternative IDs must be used. | aTRZohPigkiVWjEdjjobzQ_V <input type="button" value="Export Server Info..."/> |
| Alt. Server ID 1 | Compound ID of network card information. | d93206aeeb14207005359432b1c15f9120774953_V |
| Alt. Server ID 2 | ID of active network card. | 000c29398f7b_V |
| U3 Smart Drive ID | The ID on the right side (if visible) can be used if you like to have a portable ID, i.e. the licenses for that ID are valid on any machine where this U3 drive is connected to. | |
| <input type="button" value="Change the checked attributes now"/> | | |

Figure 7: OLicense-Server configuration window

To change the OLicense-Server settings, check the checkbox in the first column of the corresponding row; enter your changes and press the change the checked attributes now button.



Your changes won't be applied, if the checkbox in the first column of the corresponding row isn't checked.

Description of rows:

| Column | Description |
|------------------------------------|---|
| <p>User & Password:</p> | <p>Name and password of the server administrator account. Retype your password into input box Confirm to avoid mistakes. All characters in input box Password and Confirm are shown as asterisks (*).</p> <p>It's recommended to choose a password which is different from your other passwords because communication between browser and OLicense-Server is not encrypted.</p> <p>The account can be deleted if the checkbox 'delete account' is checked. After deletion the forms can be accessed without password.</p> |
| <p>Log file:</p> | <p>Enable or disable logging. Select Yes and enter a valid path for the log file directory to enable logging. The log file will contain information about license utilization and possible error messages. OLicense-Server will create a separate log file for each month.</p> <p>Default setting is to write log files in the installation directory of OLicense-Server.</p> |

| Column | Description |
|--------------------------------|---|
| Alert E-Mail: | <p>Enter your SMTP-Server's name or address and the mail address of a person who should receive an e-mail before licenses will elapse.</p> <p>If you do not know your SMTP server's name or address, please check your browser settings. For more information see chapter Fehler! Verweisquelle konnte nicht gefunden werden. Fehler! Verweisquelle konnte nicht gefunden werden. on page Fehler! Textmarke nicht definiert..</p> |
| Pay-Per-Use File path: | <p>This information is only relevant, if pay-per-use licenses are in use. Enter the name of the directory where you want OLicense-Server to save the pay-per-use log files. OLicense-Server will create a separate log file for each month.</p> <p>Default setting is the installation directory of OLicense-Server.</p> |
| Check In/Out Server | <p>The preferred server (URL) from whom this server is intended to check in/out licenses. You should always type the full name of the URL including the port name. For instance <code>http://www.xxx.de:80</code></p> |
| Preferred Server ID | <p>The preferred server ID for Floating and PayPerUse licenses. It is based on some hardware and software informations from your system. If it is not visible the hardware manufacturer has lodged too less information, in this case you have to use one of the alternative IDs.</p> |
| Alternative Server-ID 1 | <p>A compound ID of NICs</p> |
| Alternative Server-ID 2 | <p>MAC of active NIC. (Inadvisable if your host computer uses variable MAC-addresses)</p> |
| U3-ID | <p>The IDs of common (U3-)USB-Memory-Sticks/Drives. As long as the stick(s) is/are plugged into the USB-slot(s) the licenses for that ID remain valid.</p> |

The line 'Preferred Server ID' has a button: Export-Server-Info. If you click it a file will be created which contains all ID information. Mail this file to your licensor when applying for a license.

11.5 Use case “License Check In/Out“

This use case is applied when single floating licenses are to be checked-in or -out from a main-server to a local host (notebook). “Daily licenses for notebooks” is a synonym for this use case. To perform this task the licenses have to be granted by the licensor for this use case. Input box “URL of the remote server” has to contain the URL of the preferred main server. You can predefine a value for this field by using the server administration dialog.

11.5.1 Checking out licenses

Pressing button Check Out Licenses opens the form shown below:



License Check Out

OLicense-Server

| OLicense-Server License Check Out | | | | | | |
|------------------------------------|--------------|---------|------------|---------|--------------------|------------|
| check out | product | version | procedures | options | max runtime (days) | max floats |
| <input type="checkbox"/> | Optimum-Test | 0-0 | | | 6 (max: 6) | 1 (max: 6) |
| Check out the checked licenses now | | | | | | |

Listed are all licenses, that may be checked out from the specified main server to be used by this server(notebook).

To check out licenses, you have to check the appropriate checkbox of column check out. If needed, modify the check-out parameters then press button check out checked licenses now. Afterwards you will receive a receipt informing you if the check-out process was successful.

The check-out parameters are limited by the licensor and possibly by the main server administrator to the values shown in brackets. You can strengthen the limitations but not extend them.

Example: If the run time is limited to 30 days you may change them to 5 days but not to 31.

A value of 0 means infinity. According to the previous example: If the value in brackets is 0 you may set the run time to 30 days. If the value is 30 you may not set the run time to 0.

After completion of the previous check-out steps, the desired quantity of floating licenses will be removed from the main server and placed to credit to the (notebook) server's account.

The reverse process (check-in) is either done manually or automatically according to the runtime set by the values in brackets.

11.5.2 Checking in licenses

Pressing button Check In Licenses opens the following form:



| OLicense-Server License Check In | | | | | |
|--|--------------|---------|------------|---------|-----------------|
| check in | product | version | procedures | options | floats (in use) |
| <input type="checkbox"/> | Optimum-Test | 0-0 | | | 1 (0) |
| <input type="button" value="Check in checked licenses now"/> | | | | | |

Listed are all licenses, that may be checked in from this (notebook) server to the specified main server.

Check the appropriate checkbox of column check in for each license that has to be checked in and press button check in checked licenses now.

Afterwards you will receive a receipt informing you if the check-in process succeeded.

You can't check in licenses that are used by other applications at the same time.

11.6 Use case „License Administration“

Pressing button Administrate Licenses opens a form to:

1. discharge (remove) licenses,
2. change the computer, on which OLicense-Servers is running,
3. disable licenses

4. verify/change check out's,
5. release user information for floating licenses.
6. view/change Group Floating license entries.

| OLicense-Server Admin Licenses | | | | | | | | | | |
|--------------------------------|--------------------------|-------------------------------------|--------------------------|----------------|---------|-------------------------|---------|--------------|-----------|---------------------------|
| change | unload | enabled | allow info | application | version | module | options | license type | check out | module |
| <input type="checkbox"/> | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | Test | 0-0 | print_id, print_user | | Demo | | Test-E84D226 |
| <input type="checkbox"/> | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | OLicense-Suite | 4-4 | core | MU | NodeLocked | | OptimumGmb: 626BE066@2 |

Change the checked licenses now

Figure 8: License administration

To change the license settings, check the checkbox in the first column of the corresponding row, enter your changes and press the Apply Changes button.



Your changes won't be applied, if the checkbox in the first column of the corresponding row isn't checked.

11.6.1 Unload licenses



Depending on the license model being used, a license that had been removed once from OLicense-Server's database cannot be reloaded again. Do only execute this operation if you really intend to remove licenses.

To remove a license from the OLicense-Server's database, check the corresponding checkbox in column *Remove*.

For all licenses, except floating and pay-per-use licenses OLicense-Server returns the following message:



Figure 9: OLicense-Server’s response after license removal (except for floating and pay-per-use licenses)

When removing a floating or pay-per-use license, a removal code will be returned which has to be shipped to the licensor. The removal code is shown and sent automatically to the e-mail address that has been set in the administration window for notification. More information is supplied on chapter 11.4 Use case “Server Administration” on page 27.

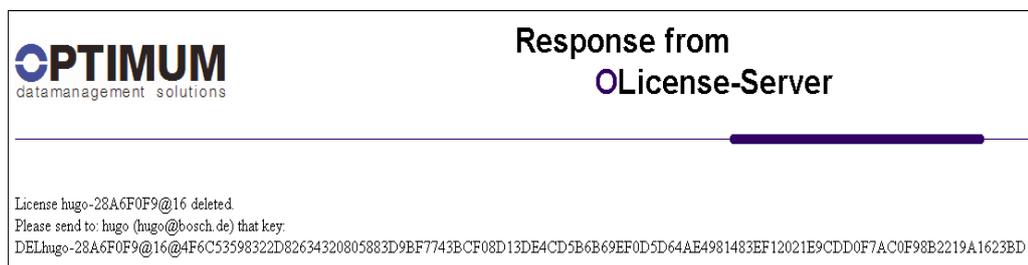


Figure 10: Removal code for floating and pay-per-use licenses

11.6.2 Changing the OLicense-Server’s hardware



Floating and pay-per-use licenses require the server key (host ID of the computer on which OLicense-Server is running). On PC’s, the server ID is derived from the hardware address (MAC address) of the first network adapter of the computer. Therefore changing the computer without taking along the network adapter invalidates these licenses.

On workstations, the hardware ID is used for that purpose.

Except in the special case where the old and the new server are PC’s and the network adapter can be taken along to the new server, all floating and pay-per-use licenses have to be regenerated by the licensor for the new server ID. For this purpose, all floating and pay-per-use licenses have to be removed from the old OLicense-Suite and the removal codes have to be sent to the licensor. Only after importing the removal codes into the

OLicense-Manager, the licenser will be able to change the server keys of the licenses and re-generate them for the new server ID.

11.6.3 Enable/disable licenses

A disabled license will not be granted by the OLicense-Server. This may be useful for testing purposes. To disable a license uncheck the checkbox in column `Enabled`. To enable a license check the checkbox in column `Enabled`.

Newly uploaded licenses are always enabled.

11.6.4 Activate/deactivate licensee feedback (supply additional information)

For floating licenses, extended feedback information of OLicense-Server can be enabled. In this case OLicense-Server returns, in conjunction with a license request, a list of those computers that have locked the available floating licenses. How this list is being followed up depends on the licensed software.

To enable feedback check the checkbox in column `allow info`.

To disable feedback uncheck the checkbox in column `allow info`.

For newly uploaded licenses this feature is disabled at first, in case the server isn't started with `-allowOccupancyInfoGlobal`, see Chapter 10.5.

11.6.5 Check out settings

If the licenser granted a license for check-out you can determine who has currently checked out licenses and you can impose additional limitations to the licensee. To do this press button `change`. See the chapter 11.7.

11.6.6 Group-Pool

In the case it is an active Group Floating license, you see here who has actually reserved a license and how long the license is reserved for a that user/host. You can see more details, edit entries and add new entries. To do this press button `view details/edit`. See the chapter 11.8

11.7 Sub use case „Change check out privileges“

Open this form by pressing button **Administrare Licenses** followed by pressing button **change in column check out license table** if it is released by the licenser.



Admin License Check Out OLicense-Server

| OLicense-Server Admin License Check Out | | | | |
|---|---------|--|--|--|
| product | version | procedures | options | license |
| Optimum-Test | 0-0 | | | OptimumGmbHRemseck-626BE066@171 |
| allowed server | | max runtime (days) | max floats | delete |
| Servers that are allowed to check out this license. {anybody} means that any server is allowed, but parameters of named servers overwrite the {anybody} settings. | | 0: unlimited; -1: server disabled (x) licenser limit | 0: up to max of current available (x) licenser limit | Delete the checked server from this list |
| {alle} | | 3 (3) | 0 (0) | |
| 0020e06c8b04:Hansi | | 6 (6) | 0 (0) | <input type="checkbox"/> |
| | | (3) | (0) | |
| Change this license now | | | | |

The third row lists common data of that license that is going to be changed with respect to its check-out privileges. From the sixth row on all server ID's are listed that may check out. Each server is represented by one row. Server ID "{all}" is an alias for arbitrary servers.

If your licenser has granted check out privileges for all servers, an additional row will be shown with an empty field where you can fill in a named server with different settings than the other servers. After committing your changes, another empty row will be displayed where you can continue in the same fashion...

The following table lists the limitations you can set. Be aware that you can only strengthen the limitations as given by the licenser but not extend them.

| Column | Description |
|--------------------|--|
| Max runtime (days) | <p>The maximum period of time for license check-out i.e. if the license isn't checked in on the main server before expiration date, the license automatically becomes invalid on expiration day and the license will be credited to the corresponding license account on the main server.</p> <p>Set this value to 0 for infinite duration</p> <p>Set this value to -1 if the specified server may not check out licenses. There is a margin of one extra day i.e. the expiration date of a license on a notebook will be extended for one day. This is done to compensate possible time intersection between license utilization on a notebook and the main server.</p> |
| Max Floats | <p>Maximum number of floating licenses that may be checked out. Set this value to 0 to enable usage of all available licenses.</p> |
| Delete | <p>The entry will be removed i.e. the upper level settings will be restored.</p> |

11.8 ***Sub use case „Admin Group Floating Licenses“***

On a Group Floating license, a user pool of a defined size is connected upstream of a floater pool. Each client who wants to lock a floating license is first directed through a user pool. If the client is already entered here with their name or computer ID, or if there is still room in the pool, the client is passed on to the floater pool. By default, the licenser defines the lifetime for the pool entries. The license is left reserved within that lifetime, regardless if the license is effectively in use or not.

| OLicense-Server Admin Group Floating License | | | | | | | |
|---|---------|---|---------|---|-----------------|---|----------------------------|
| application | version | module | options | module id | pool size | pool min lifetime | deactivation blocking time |
| Optimum-Test | 0-0 | Mod002 | | OptimumGmbH-34e90531@opt157239 | 5 (3 available) | 5 minutes | 1 minutes |
| client ID | | last usage | | status | | client lock | |
| Clients must be on this list to take a floating license | | Last time the client used the license [UTC] | | Client activity status | | When checked, client will never automatically become inactive | |
| 0815 [?@?] | | 2013-10-30 15:20:45 | | inactive <input type="button" value="Reassign"/> <input type="button" value="Deactivate"/> <input type="button" value="Deactivate and Blacklist"/> | | <input type="checkbox"/> | |
| 4711 [?@?] | | 2013-11-05 12:31:44 | | active <input type="button" value="Deactivate"/> <input type="button" value="Deactivate and Blacklist"/> | | <input type="checkbox"/> | |
| <input type="button" value="Change client locks now"/> | | | | | | | |
| add client IDs (comma separated list) | | deactivated client IDs (read only) | | automatic activation | | blacklist (comma separated list) | |
| [Host IDs] <input type="text"/> | | <input type="text"/> | | Automatic client activation is enabled , i.e. the server will automatically add a new client to the pool if a slot is available. <input type="button" value="Disable"/> | | [Host IDs] <input type="text"/> | |
| <input type="button" value="Add IDs now"/> | | | | | | <input type="button" value="Change IDs now"/> | |

Figure 11: Admin Group Floating License

The first row shows information about the license, especially the pool size, how long the entries will be allocated and how long a deleted entry will be blocked against reassigning. This data is specified by the licenser.

It follows a four column table with the pool entries:

| Column | Description |
|-------------|--|
| client-ID | User names or host IDs, according to the licensers presetting |
| last usage | Time when the client used the license at last |
| status | Active: The client has or had used the license and it is still reserved. The reservation duration results in the last usage plus the lifetime. Inactive: The validation is expired, the slot is free. |
| Client lock | If the checkbox is checked, the entry remains reserved forever. The button: <i>‘Change client locks now’</i> saves the changes. |

Active or inactive entries can be deactivated with the button: **‘deactivate’**. On deactivation the pool entry is freed immediately and its ID is put to the deactivation list.



Note: A currently used license by this entry will not be interrupted.

Entries on the deactivation list cannot be reassigned to the pool within the blocking time. The blocking time is defined by the licensor.

Active and inactive entries can be deactivated and blacklisted at the same time by clicking the button: '**Deactivate and Blacklist**'. Entries on the blacklist are always blocked from requesting a license.

Inactive entries can be reactivated with the button: '**Reactivate**'. A reactivated entry is subject to the pool lifetime.

Another four column table follows the pool table:

| Column | Description |
|------------------------|--|
| Add client IDs | Add user names or host IDs (depending on the presets of the licensor) manually to the pool. |
| Deactivated client IDs | Deactivated list (read only). Entries on that list cannot be reassigned to the pool. |
| Automatic activation | Enabled: Default. The pool is automatically filled by requesting clients. Disabled: Entries can only be added manually through this form. |
| Blacklist | See chapter: 11.10 |

11.9 Use case „Floating License Reservation“

In general a floating license permits a predefined (as defined in the license) quantity of simultaneous utilizations of the licensed software on arbitrary computers and arbitrary users. Despite of that, it might be reasonable to restrict or reserve the usage of a license to a particular group of users or computers. This can be accomplished by clicking button Reserve Floating Hosts. The following form will be opened:

| change | application | version | module | options | module id | licensee | client IDs (comma separated list of host and user IDs) | floater |
|--------------------------|-------------|---------|--------|---------|--------------------------|----------|--|---------|
| <input type="checkbox"/> | app1 | 1-1 | base | | Optimum-24b2ae9f@opt2193 | Du | dick,harry | 2 |

access limit per ID (0: unlimited)
1

Change the checked licenses now

Figure 12: Floating license reservation



If the field client IDs is filled out by your licenser you cannot make changes.

To restrict the usage of a floating license to a limited group of computers, set the ID's of those users/computers separated with comma in the corresponding row at column client IDs. You may append an alias to each ID if it is separated by a colon (:), e.g.:

```
08154711:myComputer, 4812ebd3:yourComputer,
hh:Hugo Hurtig
```

Floating licenses with column client IDs left empty may be allocated from any computer in the network.

You may add an access limit for all entered IDs. This limit describes how much licences each several uUser/host may lock. '0' means unlimited.

11.9.1 Examples for reservation

One Floating pool shall be available unlimited and solely to the users Hans, Emma and Luise:

```
IDs:      Hans,Emma,Luise
Limit :   0
```

One Floating pool shall be available to all users but Hans and Emma shall have access to one license at any time :

IDs: Hans,Emma

Limit : 1

(Floating pool size must be greater than 2 ! {Pool > Limit * NumberOfIDs})

You have 3 Floating pools. One pool of 4 (A) shall be available exclusively to Fred and Bob. One pool of 2 (B) shall be available exclusively to John, Bill, Dick and Harry. One pool of 6 (C) shall be available to all users:

A: IDs: Fred, Bob

Limit : 2

B : IDs : John, Bill, Dick, Harry

Limit: 1

C: IDs:

Limit: 0

Fred and Bob have access to ever 2 licenses from A. For more licenses they must switch to C if there is any license free.

John, Bill, Dick and Harry have access to ever 1 of 2 licenses. Do e.g. John and Dick currently using one license from B, all 4 must switch to C if they need more licenses.

11.10 Use case "Group Floating Blacklist"

Via this form, named users/hosts can be excluded from the use of a Group Floating license.

The column 'Client ID type' indicates whether the pool is operating with user IDs or with host IDs. According to this, only user IDs or host IDs can be excluded per license.

Declare the IDs as shown in the group pool column of the license administration, exclusive all clamped terms. A comment can be added with ':', e.g.: 0815cdef:MyHost:MAC

12 High availability server system

For all license models, except for pay-per-use and floating licenses, you may easily establish a high availability server system by setting up several servers on different hosts, supplying each server with the same licenses. You may specify up to three servers in your client application that can be checked automatically for available licenses. Even if two of three servers should fail the application will work anyway.

Pay-per-use licenses work in the same fashion, however the licensor has to create an own license for each server and he has to use the log-files of all servers for payoff.

Floating licenses require to setup a special highest availability server system, which is available in two main variations. First variation can be achieved by combining different servers with one or more OLicense-Proxy-Servers. This framework can be again established with different variations. Each framework consists of at least three standard OLicense-Server (called server), each of it with its own database files and at least one OLicense-Server running in proxy mode (called proxy).

Server and proxies evidently are web services that can be requested in the intranet/internet without restrictions.

The proxy is responsible for hiding the complete server framework from the licensed client application as well as administration by using a browser. The client application can access only the URL's of the proxies.

The proxy's job is to select, control and synchronize the servers. The proxy isn't bound to a particular host but the servers are i.e. to be able to get a license from the licensor the proxy must be provided with the server ID's of all servers that are part of the license framework.



Caution! The server IDs cannot be requested through the proxy! You have to contact each server separately.

Before requesting a license from the licensor, you have to define your server ID's and the number of servers to be used in your framework.

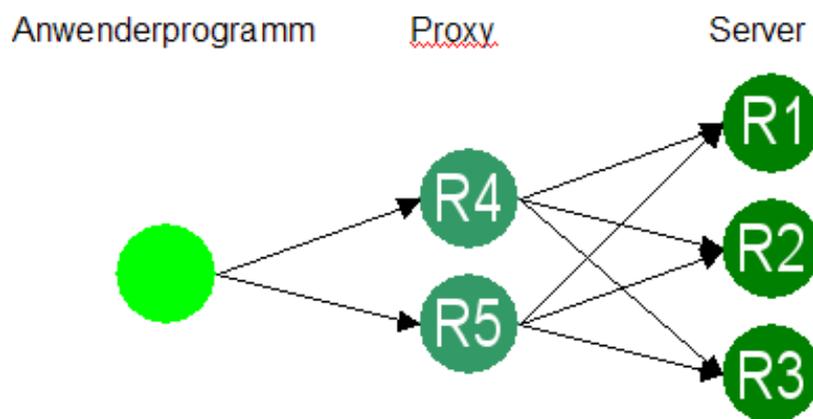
Evidently you can also use proxies for the other license models.

12.1 Server system configurations

Depending on the number of computers, we will propose three variants (*optimal, minimal and critical*) in the following sections:

12.1.1 Optimal

This requires five computers:



computer 1-3 each hosts a server, computer 4-5 each hosts a proxy.

In this configuration, the failure of one server **and** one proxy can be compensated without losing functionality.

This system has a moderate redundancy with outstanding performance.

12.1.2 Minimal

This requires three computers: computer 1-3 each hosts a server, computer 1-2 each hosts a proxy.

In this configuration, the failure of one server **or** one proxy can be compensated without losing functionality.

This system has a single redundancy with good performance. Apply this configuration only, if you haven't enough computers available. See the annotation in section starting/stopping proxies.

12.1.3 Critical

This requires a total of eight computers: computer 1-5 each hosts a server, computer 6-8 each hosts a proxy. In this configuration, the failure of two

servers **and** two proxies can be compensated without losing functionality.

This system has high redundancy with sufficient performance. You can increase redundancy at your own preference at the cost of decreasing performance. Conceivably are for instance twelve computers with nine servers and three proxies. This configuration withstands the failure of four servers and two proxies. Proxy failure is not critical, because if three proxy hosts are broken you may start three new proxies on different hosts. In this case you should inform the licensees about the new addresses.

12.1.4 Mixed Mode

In a server system, only multi-server-floating licenses are synchronized i.e. all other license models in the framework (including single-server-floating licenses) remain unchanged on the server. Therefore a licensed application can use the licenses either directly via server or via proxy.

Also conceivable is to combine servers that are located in different departments of your company and supplied with single licenses to a highly availability server system by using proxies. This facilitates multi-server-license operation.

12.2 *Starting/Stopping of servers in a server system*

No difference compared to a single server.

If required, you can start the server with parameter `-proxyOnly` which enforces proxy control. In this case you cannot use mixed mode operation.

12.3 *Starting/Stopping of proxies in a server system*

Through the Controller frontend or from the command line:

```
olicenseserver [-port <port number>] -servers  
<server1>,  
<server2>,<server3>[,<serverN>,...]
```

If parameter `-port` isn't specified port 80 will be used as default.

Parameter `-server` expects a list of all servers that are members of your server system; each entry has to be separated by a comma.

For instance:

```
olicenseserver -servers  
www.hugo.de,192.168.5.10:8080,ernie.local.de:8090
```

If the proxy-server is started for the first time, it registers itself as a service and it will be available automatically after each reboot. If you intend to change the server parameters or other parameters you must remove the proxy-service at first (`olicenseserver -uninstallService`) and then restart it with the modified parameters.



If you use a browser to administrate the proxy and you receive a message box for more than 60 seconds stating that the server is in synchronization mode this may be caused by the inability to find the specified servers. We advise to start the proxy temporarily with parameter `-logFile <bla.log>` to find the cause. One possible cause could be that the server names are not specified full qualified.

Annotation for operation in minimal configuration (see above):

If you intend to use a server and a proxy as service on the same host, ensure that their service names are different. This can be accomplished by renaming the server binary and the INI file at first:

change `olicenseserver.exe` to `olicenseproxyserver.exe` (do. for the INI file)

afterwards you can start in a similar way as described before:

```
olicenseproxyserver [-port <port>] -servers  
<server1>,...
```

Additional parameters (as described) for proxies and servers are

```
-suspendService, -resumeService, -stopService,  
-lang <language ID>
```

12.4 **Second High-Availability Variant**

All status information is constantly mirrored in variant 1. The advantage of this is that if a server fails, the remaining servers will be provided with all the previous knowledge required to continue operating the existing license locks after they have been synchronized.

Such a system naturally loses performance as a result of the mirroring process. In borderline cases, if the server needs to operate a very large number of modules from a high number of users at very short time intervals, the delays can be substantial and, depending on the programmed tolerance of the client, the latter can lose the license (possibly even permanently).

If this critical status is observed or foreseeable, the configuration according to variant 2 is recommended.

In this variant, 3 servers are installed on 3 computers without proxies. A maximum of one server may fail. Each server is configured with the -fallback... parameters such that separate communication between the servers themselves is possible.

The general syntax for each server is as follows:

```
olicensserver [-port <port number>] -fallbackMyself  
<my tcp>[:<my port>] -fallbackServer1 <its tcp>[:<its  
port>]>] -fallbackServer2 <its tcp>[:<its port>]
```

The entry for “tcp” must be made in numeric form, i.e. DNS names are not permitted.

E.g.:

PC 1 on 192.168.0.1

```
olicensserver -port 8080 -fallbackMyself  
192.168.0.1:9090 -fallbackServer1 192.168.0.2:9090  
-fallbackServer2 192.168.0.3:9091
```

PC 2 on 192.168.0.2

```
olicenseServer -port 8080 -fallbackMyself  
192.168.0.2:9090 -fallbackServer1 192.168.0.3:9091  
-fallbackServer2 192.168.0.1:9090
```

PC 3 on 192.168.0.3

```
olicenseServer -port 8085 -fallbackMyself  
192.168.0.3:9091 -fallbackServer1 192.168.0.1:9090  
-fallbackServer2 192.168.0.2:9090
```



The TCP paths between the servers themselves must be stable. Frequent interruptions will lead to the servers being permanently blocked, which will necessitate a reboot.

A license that must be obtained from the manufacturer requires the server ID of all 3 servers. This type of license also functions in variant 1 with 3 servers.

The client application will be informed of all 3 servers according to the manufacturer's guidelines.

Via the start parameters, as standard: -OLicenseServer

E.g.:

```
myApp -OLicenseServer  
192.168.0.1:8080,192.168.0.2:8080,192.168.0.3:8085
```

(DNS names may also be entered here.)

Advantages of variant 2:

- Almost the same performance as a single server system
- Lower resource consumption
- No proxies

Disadvantages:

- If servers fail, clients can lose their current license locks and need to go through the assignment procedure again.

13 Backup and Restore

13.1 Administration of database files

By default, the server saves the uploaded license information in an encrypted, object oriented database, consisting of three files located in its installation directory (default: C:/Program Files/optimum/olicense).

The default file names are:

1. OlicenseServerDB.odb:
Contains all information entered or uploaded into the OLicense-Server.
2. OlicenseServerDB.odx:
Index file for OLicenseServerDB.odb.
3. OlicenseServerDB.ini:
Contains the configuration settings for OLicenseServerDB.odb.

By default, each *.odb file has a minimum size of 100 kB that is increased in steps of 100 kB if required until it reaches 1 GB. A single license allocates an average size of about 5 kB, meaning the default configuration can handle about 200.000 licenses.

13.2 Backup database files

For backup, all *.odb files as well as all <name>.ini and <name>.odx have to be saved.



As long as the OLicense-Server is running, the database files are locked. Stop the OLicense-Server as described in chapter 10.2 on page 16.

13.3 Restoring database files

To restore the backup database files, stop OLicense-Server as described in chapter 10.2 on page 16. Replace the database files by your backup files. For security reasons, we recommend to remove the old database files completely from the directory before restoring the backup files. Afterwards restart the OLicense-Server as described in chapter 10, *Starting and Stopping Olicense-Server* on page 14.

13.4 Reinstallation of OLicense-Server

Case 1: Reinstallation on the same computer:

1. Make a backup of your database files as described in chapter 13.2, *Backup database files* on page 47.
2. Reinstall the OLicense-Server from the CD.
3. Restore the database files as described in chapter 13.3 *Restoring database files* on page 39.
4. Restart OLicense-Server as described in chapter 10, *Starting and Stopping Olicense-Server* on page 14.

Case 2: Reinstallation on a different computer:

If you have only licenses that don't depend on the server-ID, you can proceed as described in case 1.

In all other cases, proceed as described in chapter 11.6.2 *Changing the OLicense-Server's hardware* on page 28.

13.5 *Backup in batch mode*

To realize automatic backup of the OLicense-Server's database files, you can write a simple batch job.

Example:

```
olicensserver -stopService  
copy ...odb ...  
copy ...ini ...  
copy ...odx ...  
olicensserver -startService
```

14 (U3)-USB-Memory-Stick

Each stick/disk is assigned an individual hardware ID to which licenses can be bound. This facilitates it to create a pluggable OLicense-Server, that can be used on different computers. This means that the licenses are only valid on that computer where the OLicense-stick is plugged in.

U3 sticks are USB sticks with extended functionality. Since 2012 sale and support is no longer available. But OLicense still supports the readout of the U3-ID.

The server recognizes all plugged sticks and shows the IDs on the admin page. Every single ID may be used for licensing.

14.1 *Specifics for Linux*

Under Linux, the rights for reading the relevant parameters are usually only accessible to the root user. If the server does not have root rights for the parameter `-runAs <user>:<group>`, it will not be able to read the stick. In this case, you will need to register the sticks to which licenses are linked in the system. Proceed as follows:

Plug the stick into the computer.

Enter the command `lsusb` to find the manufacturer ID of the stick. The command output contains several lines of the following type:

```
BUS 001 Device 001: ID <XXXX>:<NNNN> <plain text description>
```

Using the plain text description, localize your stick and note the two number blocks by ID. **XXXX** stands for the vendor ID and **NNNN** for the product ID.

Switch to the directory `/etc/udev/rules.d` and create a file with the name `80-OLicense.rules`. (If this directory contains different name syntax, please adhere to this syntax).

Open the file in a text editor and for each licensed stick write a line in the following format:

```
SYSFS{idVendor}=="XXXX", SYSFS{idProduct}=="NNNN",  
MODE="0660", GROUP="GGGG"
```

Replace the Xs and Ns with the IDs you have read. Replace the Gs with the group name that you entered in the server with `-runAs`
<UUUU>:<GGGG>.

Now remount the stick.

(Note: The procedure described above refers to SuSE Linux 11. It is theoretical possible that other distributions handle this in a different manner. Please check your system documentation).

15 Serverstatistics

Please use external applications for that purpose.

(see e.g.: x-formation.com)

16 olixtool – the command line tool

olixtool is a non graphical application which offers mainly the information functions from the OLicense-Support-API. It's purpose is to retrieve information from the client- but mainly the server-side by hand or batch mode.

Calling it without any parameter will show a short help.

The following optional parameters are available:

| Parameter | Value | Description |
|---------------------------|--|--|
| Global | | |
| -OLicenseServer -sv | <serveraddress>[:<Port>] | Address of the server to retrieve information from. Default: <i>local,localhost</i> |
| -OLicenseHttpProxy -pr | [<user>:<pw>@]<serveraddress>[:<Port>] | Address and logindata of a proxyserver if existent. Default: <i>No Proxy</i> |
| Output | | |
| -ListOutput -lst | | The result is printed as simplified, single-line list without comments. Default: <i>human readable</i> |
| -XmlOutput -xml | | The result is printed as complete XML structure. Default: |

| | | <i>human readable</i> |
|-----------------------------|---|--|
| Licensestatus-Filter | | |
| -LicenserKey -lk | <License-root-keyname of the application> | Rootname of the licenser key, see column: <i>Key</i> (without key number) in the license report of the server. |
| -AppName -an | <applicationname> | Name of the application, see column: <i>Application</i> in the license report of the server. |
| -AppVersion -av | <versionnumber> | Versionnumber of the application |
| -ModuleName -mn | <modulename> | Name of the application module, see column: <i>Module</i> in the license report of the server. |
| -Licenser -lr | <name of licenser> | See column: <i>Licenser</i> in the license report of the server. |
| -Licensee -le | <name of licensee> | see column: <i>Licensee</i> in the license report of the server. |
| -LockedOnly -lck | | Lists only Floating licenses which are currently locked by users. |
| -ClientIDs -cid | [<ID1>[,<ID2>,...]] | Lists only licenses which match the given IDs. If no IDs are given the local ID is used, i.e. |

| | | |
|---|--|--|
| | | an ID which a client would have. |
| Host-ID | | |
| -HostIDs -hid [-IdLockedOnly -ido] | [-user <OLicenseAdmin> - pw <password>] | Reports all server- and client IDs. If the server is protected by an account the loginname and password must be given. Is the additional parameter – IdLockedOnly present, only the client Ids are reported. |
| Version | | |
| -ServerVersion -v | | Reports the version string of the addressed server.. |
| Import | | |
| -Import -imp | <filename> | Imports the given file in the server. |

Examples:

Who has currently locked which modules of application *foo*:

```
olixtool -sv bla.fasel.de:8080 -an foo -lck
```



(Caution! To retrieve user information the server administrator has to enable this information before, siehe Chapter: 11.6.4, Page 34)

Which modules does application *foo* have:

```
olixtool -sv localhost -an foo -lst
```

Which prospective host IDs can I send to my licenser:

```
olixtool -hid -xml > myIds.txt
```

17 Appendix: Messages

17.1 Transaction errors

17.1.1 Communication failure

| Sender | No. | Text | Description |
|--------|-----|----------------------------------|--|
| OIComm | 408 | Receive timeout | OLicense-API didn't receive a reply within 10 seconds. This may be caused by a blocked server host or the License-Server process was suspended etc. |
| OIComm | 450 | Receive out of memory | Reply has too much data. Possibly the reply doesn't belong to OLicense-Server. |
| OIComm | 451 | Request cannot open socket | Socket error of client. This may be caused by insufficient resources or TCP/IP protocol isn't supported. |
| OIComm | 452 | Request cannot find hostname | Unable to get the hostname of the computer on which the licensed software is running. This may be caused by wrong operating system settings. |
| OIComm | 453 | Request cannot reuse clientport | Unable to reuse client port. Port is probably being used by another program. Choose another port or disable client port. |
| OIComm | 454 | Request cannot connect to target | The connection to OLicense-Server failed. Check the address settings of the computer on which OLicense-Server is running. Check proxy and firewall settings. |
| OIComm | 455 | Request cannot send to target | An error occurred during transmission of data to the OLicense-Server. Possible cause: broken down network. |
| OIComm | 550 | Receive error: %d | Network subsystem error. The error code indicates the type of error. Lookup in the operating system error table to get more information. |

17.1.2 Protocol error

| Sender | No. | Text | Description |
|--------|-----|--|--|
| OISoap | 560 | Message to short; invalid HTTP | Reply has wrong format. This message cannot result from a OLicense-Server. |
| OISoap | 561 | Message invalid HTTP | Reply is not HTTP. This message cannot result from a OLicense-Server. |
| OISoap | 562 | Message cannot scan HTTP response code | Reply has no response code. This message cannot result from a OLicense-Server. |
| OISoap | 601 | Message response data error/Missing | Reply has no data. This message cannot result from a OLicense-Server. |
| OISoap | 602 | Message transaction code error | Wrong transaction code. Reply cannot result from a send message. Probably caused by a network error. |
| OISoap | 603 | Message transaction code missing | Reply with transaction code missing. This message cannot result from OLicense-Server. |

17.1.3 OLicense-Server errors

| Sender | No. | Text | Description |
|----------|-----|---|--|
| OIServer | 700 | Transaction number missing | Sent message has no transaction number. This message cannot result from a OLicense-Server. |
| OIServer | 703 | No floating license free | All floating licenses are being used. |
| OIServer | 704 | Server can't write Pay-per-Use time | OLicense-Server is unable to write log-file for pay-per-use licenses. Check the settings in the administration form of OLicense-Server. |
| OIServer | 705 | No licenses available for user/host | No license available for this user or host. Check if user has a valid named-user- or floating-license or if a node locked license is available for this host. Check if user name respective host name are correct. |
| OIServer | 706 | No licenses available for product | No valid license for this product. Check if product has a license or if product name is correct. |
| OIServer | 707 | Invalid date or key mismatch in request | Wrong key or invalid date in request. Check the date settings of server and client they must be equal. |
| OIServer | 708 | Invalid license (key missing) | Check the license file into OLicense-Server again. |
| OIServer | 709 | No license available for request key | Request a new license from the manufacturer. |
| OIServer | 710 | <...> | Server/Database error. To resolve this you may restore the database files of OLicense-Server. |
| OIServer | 711 | Group pool fully occupied | The Group pool is fully occupied or the user/host has been excluded from usage |

17.2 *Transaction ok*

| Sender | No. | Text | Description |
|----------|-----|------|---|
| OIServer | 200 | Ok | After successful license check OLicense-Server has unlocked the licensed application. |

17.3 *Error (exception) caused by erroneous programming with OLicense API*

| Sender | No. | Text | Description |
|-----------|-----|---|---|
| OILicense | 100 | Object not initialized | OLicense API wasn't initialised. |
| OILicense | 101 | Unbalanced lock count | Missing log-off of licenses |
| OILicense | 102 | Cannot change user name while locks are active | Wrong point of time for usage of this function. |
| OILicense | 103 | Cannot change server address while locks are active | Wrong point of time for usage of this function. |
| OILicense | 104 | Object already initialized | Multiple initialization of OLicense API |

18 Further information

Our postal address is:

Optimum GmbH
Hirschstr. 12-14
D-76133 Karlsruhe

The e-mail address of our support is:

support@optimum-gmbh.de

Optimum GmbH

to get further information about Optimum and our products just visit our homepage at:

<http://www.optimum-gmbh.com>

We offer products and services regarding following topics:

- Data- and workflow-management
- Development of machine vision applications
- Development of client-server applications
- License management and software protection