



TASKING License Management

Copyright © 2015 Altium BV.

All rights reserved. You are permitted to print this document provided that (1) the use of such is for personal use only and will not be copied or posted on any network computer or broadcast in any media, and (2) no modifications of the document is made. Unauthorized duplication, in whole or part, of this document by any means, mechanical or electronic, including translation into another language, except for brief excerpts in published reviews, is prohibited without the express written permission of Altium BV. Unauthorized duplication of this work may also be prohibited by local statute. Violators may be subject to both criminal and civil penalties, including fines and/or imprisonment. Altium, TASKING, and their respective logos are trademarks or registered trademarks of Altium Limited or its subsidiaries. All other registered or unregistered trademarks referenced herein are the property of their respective owners and no trademark rights to the same are claimed.

Table of Contents

Manual Purpose and Structure	v
1. Licensing Overview	1
1.1. Licensing Terminology	1
1.2. License Types	2
2. Licensing TASKING Products	5
2.1. Obtaining a License	5
2.2. Frequently Asked Questions (FAQ)	6
2.3. Installing a License	6
2.3.1. Configure the Firewall in your Network	6
2.3.2. Installing Server Based Licenses (Floating or Node-Locked)	6
2.3.3. Installing Client Based Licenses (Node-Locked)	7
2.4. License Files Created During Installation	8
2.4.1. Licensing Options File	9
2.4.2. License File	11
3. Local License Server	13
3.1. Overview	13
3.2. Installing a Local License Server	14
3.2.1. Start the Local License Server as a Service	15
3.2.2. Start the Local License Server from the Command Line	16
3.2.3. Obtaining a License for a Local License Server	16
3.3. Local License Server Options	16
3.4. Redundant Master - Slave Configuration	18
3.5. Creating a Local License Server License File	20
4. Administering Licenses	23
4.1. Monitoring the Status of the Local License Server	23
4.2. Transferring a License	24
4.3. Joining Licenses	25
4.4. Bulk Installation	26
4.5. Restricting Access to Licenses	26

Manual Purpose and Structure

Manual Purpose

You should read this manual if you:

- install a licensed TASKING product and need more information about licensing and installation than what is described in the Getting Started manual of the TASKING product
- are upgrading or otherwise modifying existing licenses
- install a local TASKING license server on a network

Manual Structure

Chapter 1, *Licensing Overview*

Contains an overview of the terminology used in this manual and describes the different license types.

Chapter 2, *Licensing TASKING Products*

Explains step-by-step how to obtain and install a license for a TASKING product. Also contains a list of options and environment variables that a TASKING product uses to check for a license. Furthermore it explains the use of a license file.

Chapter 3, *Local License Server*

Explains how to install and configure a local license server.

Chapter 4, *Administering Licenses*

Explains how to monitor the status of the local license server. Also explains how to borrow, join or bulk install licenses and control access to licenses.

Related Publications

For details about installing a TASKING product, see the Getting Started manual delivered with the product.

Chapter 1. Licensing Overview

1.1. Licensing Terminology

TASKING products are protected with TASKING license management software. To use a TASKING product, you must install that product and install a license.

The following terminology is used to describe the licensing concepts and software components (in alphabetical order).

Access file

An access file is a file used by a local license server with rules to deny, permit, block and bind users and systems to licenses.

Client

A client is a TASKING product for which a license is needed. For example, a C compiler.

Computer ID

The computer ID is an alphanumeric value that uniquely identifies your computer. The Computer ID is required to identify the computer for a node-locked license.

License

A license is the right to use a product. During the execution of a TASKING product (client), the client verifies whether you have the right to use it. Therefore, the client sends the license key to the license server. If you are allowed to use the client the license server sends a license back to the client. If you use a node-locked client based license the license is not retrieved from a license server but is read from a license file stored on your computer.

License key

The license key is a unique key (in the format xxxx-xxxx-xxxx-xxxx) provided to you by Altium. The license key is associated with the product you ordered, it identifies that you have the right to use the product. During the installation of your product you need to enter this key.

License server

A license server is the process that handles the license requests from client processes. A license server can be a remote license server on the internet or a local license server on your company network.

In this manual "Local TASKING License Server" and "Local License Server" are used as synonyms. "Remote License Server" is used as a synonym for the "Remote TASKING License Server".

Security ID

The security ID is an alphanumeric value that uniquely identifies the computer that runs a local TASKING license server.

1.2. License Types

The following license types can be ordered from Altium.

Node-locked license

A node-locked license locks the software to one specific computer so you can use the product on that particular computer only.

For information about installing a node-locked license see [Section 2.3.2, Installing Server Based Licenses \(Floating or Node-Locked\)](#) and [Section 2.3.3, Installing Client Based Licenses \(Node-Locked\)](#).

Floating license

A floating license is a license located on a license server and can be used by multiple users on the network. Floating licenses allow you to share licenses among a group of users up to the number of users (seats) specified in the license.

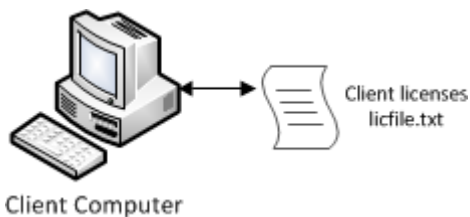
For example, suppose 50 developers may use a client but only ten clients are running at any given time. In this scenario, you only require a ten seats floating license. When all ten licenses are in use, no other client instance can be used.

For information about installing a floating license see [Section 2.3.2, Installing Server Based Licenses \(Floating or Node-Locked\)](#).

License service types

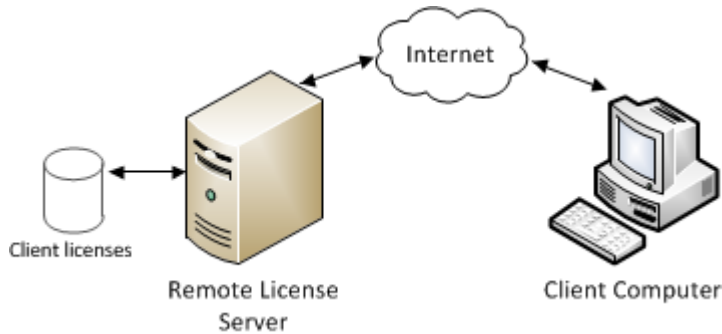
The license service type specifies the process used to validate the license. The following types are possible:

- **Client based** (also known as 'standalone'). The license is serviced by the client. All information necessary to service the license is available on the computer that executes the TASKING product. This license service type is available for node-locked licenses only.

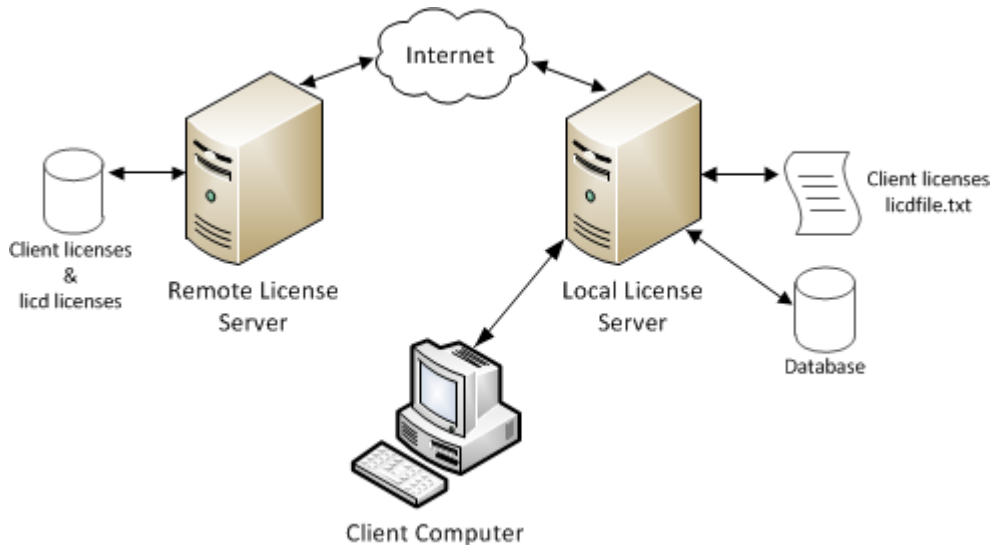


- **Server based** (also known as 'network based'). The license is serviced by a separate license server program that runs either on your companies' network or runs in the cloud. This license service type is available for both node-locked licenses and floating licenses.

Licenses can be serviced by a cloud based license server called "**Remote TASKING License Server**". This is a license server that is operated by TASKING.



Alternatively, you can install a license server program on your local network. Such a server is called a "**Local TASKING License Server**". You have to configure such a license server yourself, see [Section 3.2, Installing a Local License Server](#).



The benefit of using the Remote TASKING License Server is that product installation and configuration is simplified.

Unless you have an IT department that is proficient with the setup and configuration of licensing systems we recommend to use the facilities offered by the Remote TASKING License Server.

TASKING License Management

Chapter 2. Licensing TASKING Products

This chapter guides you through the process of licensing a TASKING product.

Installation advice when using a local license server

During the installation of a TASKING product you can specify to use a local license server. If you do so, the installation program asks for the server name and port number of the local license server. If you do not know this information, you can simplify the installation if you first install and run the local license server before you install the TASKING product. See [Section 3.2, *Installing a Local License Server*](#).

2.1. Obtaining a License

You need a license key when you install a TASKING product on a computer. If you have not received such a license key follow the steps below to obtain one. Otherwise, you cannot install the software.

Obtaining a server based license (floating or node-locked)

- Order a TASKING product from Altium or one of its distributors.

A license key will be sent to you by email or on paper.

If your node-locked server based license is not yet bound to a specific computer ID, the license server binds the license to the computer that first uses the license.

Obtaining a client based license (node-locked)

To use a TASKING product on one particular computer with a license file, Altium needs to know the computer ID that uniquely identifies your computer. You can do this with the **getcid** program that is available on the TASKING website. The detailed steps are explained below.

1. Download the **getcid** program from <http://www.tasking.com/support/tlm/>.
2. Execute the **getcid** program on the computer on which you want to use a TASKING product. The tool has no options. For example,

```
C:\Tasking\getcid  
Computer ID: 5Dzm-L9+Z-WFbO-aMkU-5Dzm-L9+Z-WFbO-aMkU-MDAy-Y2Zm
```

The computer ID is displayed on your screen.

3. Order a TASKING product from Altium or one of its distributors and supply the computer ID.

A license key and a license file will be sent to you by email or on paper.

When you have received your TASKING product, you are now ready to install it.

2.2. Frequently Asked Questions (FAQ)

If you have questions or encounter problems you can check the support page on the TASKING website.

<http://www.tasking.com/support/tlm/faq.shtml>

This page contains answers to questions for the TASKING license management system TLM.

If your question is not there, please contact your nearest Altium Sales & Support Center or Value Added Reseller.

2.3. Installing a License

The license setup procedure is done by the installation program.

If the installation program can access the internet then you only need the licence key. Given the license key the installation program retrieves all required information from the remote TASKING license server. The install program sends the license key and the computer ID of the computer on which the installation program is running to the remote TASKING license server, no other data is transmitted.

If the installation program cannot access the internet the installation program asks you to enter the required information by hand. If you install a node-locked client based license you should have the license file at hand (see [Section 2.1, Obtaining a License](#)).

Floating licenses are always server based and node-locked licenses can be server based. All server based licenses are installed using the same procedure.

2.3.1. Configure the Firewall in your Network

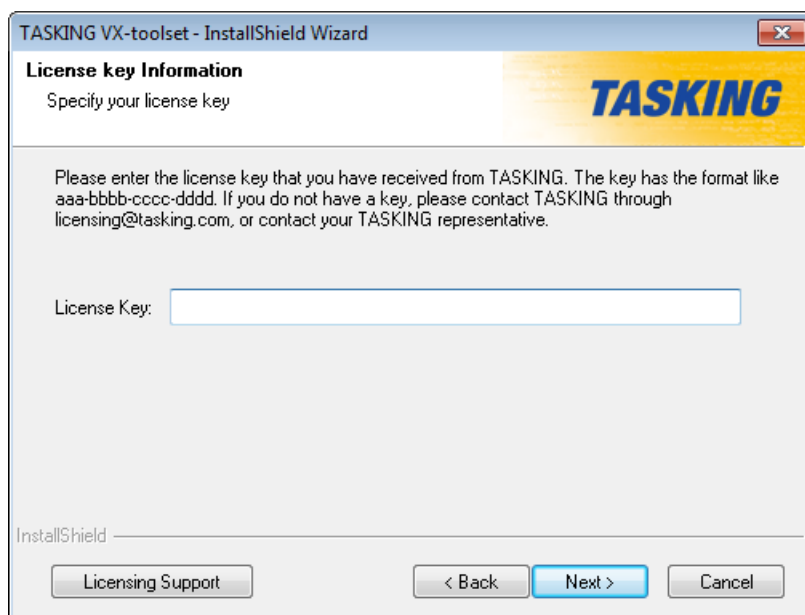
For using the TASKING license servers the TASKING license manager tries to connect to the Remote TASKING servers `lic1.tasking.com` .. `lic4.tasking.com` at the TCP ports 8080, 8936 or 80. Make sure that the firewall in your network is transparently enabled for one of these ports.

2.3.2. Installing Server Based Licenses (Floating or Node-Locked)

If you do not have received your license key, read [Section 2.1, Obtaining a License](#) before you continue.

1. If you want to use a local license server, first install and run the local license server before you continue with step 2. See [Section 3.2, Installing a Local License Server](#).
2. Install the TASKING product and follow the instruction that appear on your screen.

The installation program asks you to enter the license information.



3. In the **License key** field enter the license key you have received from Altium and click **Next** to continue.

*The installation program tries to retrieve the license information from a remote TASKING license server. Wait until the license information is retrieved. If the license information is retrieved successfully subsequent dialogs are already filled-in and you only have to confirm the contents of the dialogs by clicking the **Next** button. If the license information is not retrieved successfully you have to enter the information by hand.*

4. Select your **License type** and click **Next** to continue.

You can find the license type in the email or paper that contains the license key.

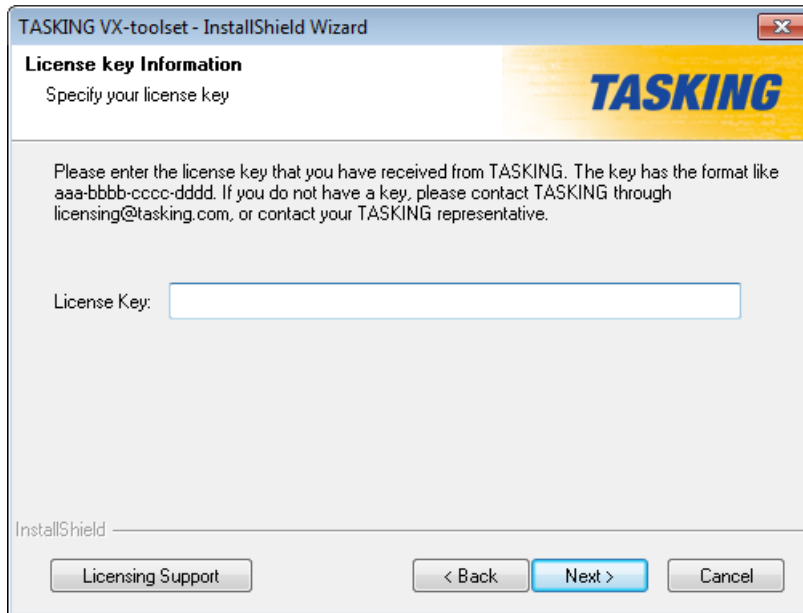
5. Select **Remote TASKING license server** to use one of the remote TASKING license servers, or select **Local TASKING license server** for a local license server. The latter requires optional software.
6. (For local license server only) specify the **Server name** and **Port number** of the local license server.
7. Click **Finish** to complete the installation.

2.3.3. Installing Client Based Licenses (Node-Locked)

If you do not have received your license key and license file, read [Section 2.1, Obtaining a License](#) before continuing.

1. Install the TASKING product and follow the instruction that appear on your screen.

The installation program asks you to enter the license information.



2. In the **License key** field enter the license key you have received from Altium and click **Next** to continue.

*The installation program tries to retrieve the license information from a remote TASKING license server. Wait until the license information is retrieved. If the license information is retrieved successfully subsequent dialogs are already filled-in and you only have to confirm the contents of the dialogs by clicking the **Next** button. If the license information is not retrieved successfully you have to enter the information by hand.*

3. Select **Node-locked client based license** and click **Next** to continue.
4. In the **License file content** field enter the contents of the license file you have received from Altium.

The license data is stored in the file `licfile.txt` in the `etc` directory of the product.

5. Click **Finish** to complete the installation.

2.4. License Files Created During Installation

The installation program stores the license key in the file `licopt.txt` in the `etc` directory of the product. See [Section 2.4.1, Licensing Options File](#) for details about this file. Additionally, for a node-locked client base license only, the installation program stores the license data in the file `licfile.txt` in the `etc` directory of the product. See [Section 2.4.2, License File](#) for more information.

2.4.1. Licensing Options File

All information needed by the client program to perform the license check is read from the options file `licopt.txt` or from environment variables. The options in the options file have precedence over the environment variables.

The options file `licopt.txt` is searched for according to the following search path, relative to the installation directory of the product:

Windows:

```

.:.\etc;..\etc;tooldir\..\etc;tooldir\..\..\etc;tooldir\..\bin;tooldir

```

UNIX:

```
./etc:./../etc:toolldir/./etc:toolldir/./../etc:toolldir/./bin:toolldir
```

Alternatively, you can specify the name and location of the options file in an environment variable:

```
TSK_OPTIONS_FILE = file
```

or

```
TSK_OPTIONS_FILE_Product-Code = file
```

or

```
TSK_OPTIONS_FILE_Product-Code-version = file
```

For example:

```
TSK_OPTIONS_FILE = myoptions.txt
```

```
TSK_OPTIONS_FILE_SW103800v5_1r1 = sw103800-opt.txt
```

The following options can be present in the license options file or as an environment variable:

Option / Environment Variable	Description	Mandatory / Optional
TSK_LICENSE_KEY_ <i>Product-Code</i> = <i>License-key</i> PRODUCT	Assign the license key <i>License-key</i> to the TASKING product with product code <i>Product-Code</i> . If the string PRODUCT is entered instead of a license key, then the client can retrieve an arbitrary license associated with the TASKING product identified by <i>Product-Code</i> from the license server. This functionality is, for example, used for joining licenses (see Section 4.3, Joining Licenses). You can find the <i>Product-Code</i> in the email or paper that contains the license key.	Mandatory

Option / Environment Variable	Description	Mandatory / Optional
TSK_LICENSE_SERVER = <i>host</i> [: <i>port</i>] [...]	Specifies the list of license servers and associated port numbers that can be contacted by a client. A client tries to connect to a server in the order specified in the list. Only if a server is offline then a connection to another (backup) server is made. The server can be a TASKING license server or a local licenser server. The default is: lic1.tasking.com:8080; lic1.tasking.com:8936; lic1.tasking.com:80; lic2.tasking.com:8080; lic2.tasking.com:8936; lic2.tasking.com:80; lic3.tasking.com:8080; lic3.tasking.com:8936; lic3.tasking.com:80; lic4.tasking.com:8080; lic4.tasking.com:8936; lic4.tasking.com:80	Optional
TSK_LICENSE_PORT = <i>port</i>	Connect to port <i>port</i> on the license server. For the remote TASKING license server you can use port 8080, 8936 or port 80 (the remote TASKING license server listens on all three ports). If you use a local licenser server, specify 9090 or the port that you used during the installation of the local license server.	Optional
TSK_LICENSE_PROXY_SERVER = <i>proxyserver</i> [: <i>port</i>]	If a proxy server is required for outgoing license requests, use this option to specify <i>proxyserver</i> as the HTTP proxy server. If you omit the <i>port</i> , the default port is 3128.	Optional
TSK_LICENSE_PROXY_PORT = <i>port</i>	Specify the TCP port number to use for a proxy server (default: 3128).	Optional
TSK_LICENSE_FILE = <i>file</i> or TSK_LICENSE_FILE_Product-Code = <i>file</i> (environment variable only)	If this option or environment variable is set, the client retrieves a license from the license file <i>file</i> instead of a remote server. Use this option when you use a node-locked client-based license. (default: <i>licfile.txt</i>). The search path is the same as for the options file. Alternatively, you can specify an absolute path. If you have several license files for different products you can use the product code in the environment variable name.	Optional

Option / Environment Variable	Description	Mandatory / Optional
TSK_LOG_VERBOSE	Log verbose, repeat for more verbose. With this option log information about a license request by a client is sent to <code>stderr</code> . This option can be useful to see what is happening when problems arise during licensing.	Optional
TSK_NET_TIMEOUT = <i>sec</i>	Set the network timeout (default: 4 seconds). This specifies the number of seconds the system is allowed to wait for a connection.	Optional
TSK_NO_ANONYMOUS	Send user name and system name with the license request. This is useful for statistics and access information of the local license server. This option is necessary when you use an access file with a local license server. See Section 4.5, Restricting Access to Licenses .	Optional

Speed up network connection

If you know that only TCP port 80, the last one in the list, is open for transparent access on your firewall, you can speed up the connection process by setting the following variable in the `licopt.txt` file:

```
# Connect to port 80
TSK_LICENSE_PORT = 80
```

or a full version:

```
TSK_LICENSE_SERVER=lic1.tasking.com:80;lic3.tasking.com:80
```

Example of an options file

```
TSK_LICENSE_KEY_SW103800 = 1283-cc0a-5e67-8ae3
TSK_LICENSE_SERVER = lic1.tasking.com:8080;lic3.tasking.com:8080
TSK_LICENSE_PROXY_SERVER = proxy.mydomain.com
TSK_LICENSE_PROXY_PORT = 8081
TSK_LOG_VERBOSE
TSK_NET_TIMEOUT = 10
```

2.4.2. License File

If you installed a node-locked client based license, a license file has been created on your computer. This file contains the license of your product.

Example of a license file:

```
#
# TASKING License File
#
# Issue date: 2015-03-10 15:35:43
```

TASKING License Management

```
#  
# License type: Node-locked client based license|Joinable  
#  
# Suitable for the computer with computer ID: 5Dzm-L9+Z-WFbO-aMkU-5Dzm-L9+Z-WFbO-aMkU-MDAy-Y2Zm  
#  
# License key: 1283-cc0a-5e67-8ae3  
#  
# Product code: SW10380005044000  
#  
SW10380005044000=71R4BfrOqSwOUF14DTfNsrmIuhHEbVNaM  
O7B/BrtrKIJ2KKsBYnnB2D216TIHtBB9EDGbyi0fDE2je6RmTv  
JdlBiiQMFQ792pZT4fU7oArmAtSVx3e6XwyysSz1VdV97Z81TJ  
PS7Pe60V14wESjWGl8Q9G5yQfe4QKAzeWexkghdCBDd8RIdq+g  
sgA==
```

Chapter 3. Local License Server

3.1. Overview

Server based licenses can be serviced by either the remote TASKING license server or a local TASKING license server. The remote license server is accessed over the Internet whereas a local license server is accessed via your office network of your companies intranet.

A remote license server has the following advantages:

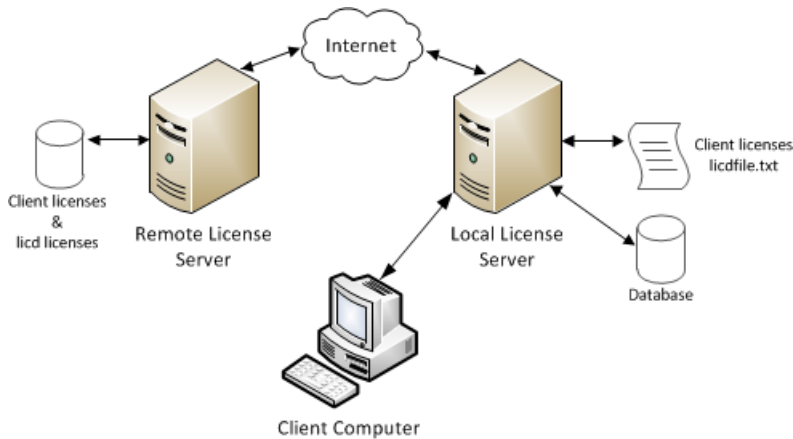
- Ease of use since you do not have to setup a local license server.
- Altium can generate a license and license key without knowing the Computer ID in advance. So, you can install the package and run it without delay, without sending the Computer ID.
- You can easily update, upgrade (e.g. from Standard to Premium Edition), add extra seats and extend your license.
- The TASKING remote license server has an uptime of 99,9%, through four servers at two different locations.

A local license server has the following advantages:

- Enables you to inspect license usage.
- Enables you to manage access to the license server and the licenses.
- Provides facilities for centralized bulk product distributions.
- Provides facilities for license management and transfer of licenses.
- Reduces internet traffic and associated delays.

The following restriction apply to a local license server:

- By default the local license server will not execute on a virtual system. To use a local license server on a virtual system you need a special license.
- Internet access is required when the local license server starts.
- Windows7 (32/64-bit), Windows8 (64-bit) or Windows Server2008R2 (64-bit) or higher.



The local license server can retrieve licenses from a license file (`licdfile.txt`) or can retrieve licenses from the remote license server, or a combination of the two. The license file is read at startup. If the license is not available in the license file, the local license server optionally retrieves the license from the remote license server. In this case the license is not available for the client on the remote license server anymore.

The local license server maintains all licenses and current license usage in a database. The licenses in the database are refreshed from the license file or from a remote server at certain intervals. When you modify the license file `licdfile.txt`, for example, to add a license, you do not need to restart the license server, since the license file is re-read automatically. When the local license server terminates, the current license usage is preserved in the database and is used again at startup. The database is locked to the security ID of the local license server's hardware. Therefore, you cannot move the database to another computer.

3.2. Installing a Local License Server

Install the local license server

1. Insert the TASKING VX-toolset CD-ROM into the CD-ROM drive.
2. If the installation program does not start automatically, browse to your CD-ROM drive and run the program **setup.exe**.

The TASKING Setup dialog box appears.

3. Select **Local TASKING License Server vx.yrz** and click on the **Install** button.
4. Follow the instructions that appear on your screen.
5. Get the security ID of the computer on which you installed the local license server by entering the following command in a command prompt:

```
licd -I
```

For more information see [Section 3.2.3, Obtaining a License for a Local License Server](#).

6. Send the security ID to Altium by email: `licadmin@tasking.com`

Altium registers the security ID on the remote license server and if you want to use a license file, Altium sends it to you.

7. If you have received a license file from Altium, store it in a file. For example: `licdfile.txt` and put it in the directory `C:\Tasking` (Windows) or the directory `etc` under the installation directory of the local license server (UNIX).

On Windows, the installation program creates the directory `C:\Tasking` if it does not exist already.

8. Add the following line to the options file `licdopt.txt`:

```
LICD_LICFILE = licdfile.txt
```

If the file `licdopt.txt` does not exist, create one and put it in the same directory as you stored the license file.

You can start the local license server as a service or from the command line.

3.2.1. Start the Local License Server as a Service

Install and start the local license server as a service (Windows only)

1. Open a command prompt as Administrator and install the local license server as a service.

```
licd --install
```

Note that without administrator rights, you cannot install a service. Note that you cannot specify other command line options, but you can set defines in `licdopt.txt`.

2. Open a command prompt as Administrator and start the local license server as a service.

```
sc start "TASKING License Service"
```

Note that without administrator rights, you cannot start a service.

How to stop and delete the TASKING License Service (Windows only)

If you want to uninstall the local license server, you must first stop and delete the service.

- Open a command prompt as Administrator and stop and delete the local license server as a service.

```
sc stop    "TASKING License Service"  
sc delete "TASKING License Service"
```

Note that without administrator rights, you cannot stop and delete a service.

3.2.2. Start the Local License Server from the Command Line

Instead of starting the local license server as a service, you can start the local license server from the command line.

1. Start the local license server:

```
licd [option]...
```

2. (Optional) When you get a security alert from a firewall, allow access to the network(s) that **licd** needs access to.

Internet access is required when the local license server starts since it is verified that the server can only run at the designated server.

You can only run one instance of the license server on a system. If you start a second one, the first one stops. This will lead to unexpected results.

3.2.3. Obtaining a License for a Local License Server

A local license server is licensed with the security ID of the computer on which you installed the local license server. When the local license server starts it contacts the remote license server with its security ID to get its license.

The security ID of a local license server is unique. You can obtain the security ID with the following command:

```
licd -I
```

If you install a new version of the local license server, this invalidates the database. However, the security ID is still the same, so the license of the local license server is still valid. After startup the local license server validates itself at the remote license server and retrieves the client data on demand from a license file or the remote license server and builds a new database.

3.3. Local License Server Options

You can start the local license server on the command line as follows:

```
licd [option]...
```

All options can be read from an options file (default `licdopt.txt`), or from the command line. If an options file is present, this file is always read. Command line options have precedence over options from the options file. The file is searched for in the following path:

Windows:

```
'\Tasking'
```

UNIX:

```
'../etc:/bin'
```

The defines in `licdopt.txt` are read by the local license server when installed as a service or when started from the command line. The option flags are available on the command line only. Note that you cannot use environment variables to set options for the local license server. The following options are available:

Option	Define in <code>licdopt.txt</code>	Description
-A <i>[access-file]</i>	<code>LICD_ACCESSFILE [= file]</code>	Use an access file with access rules (default: <code>licdaccess.txt</code>). See Section 4.5, Restricting Access to Licenses .
-f <i>[license-file]</i>	<code>LICD_LICFILE [= file]</code>	Use a license file instead of a remote license server (default: <code>licdfile.txt</code>). Search path: Windows: <code>'..\Tasking'</code> UNIX: <code>'../etc:/bin'</code>
-h <i>hostname[:port]</i>	<code>LICD_REM_SERVER = host[:port]</code>	Use the remote license server for LicD validation and if requested, client licenses (default: <code>lic3.tasking.com:8080</code>). Possible ports you can use are 8080, 8936 or 80.
-I		Show Security ID of the computer that executes the local license server.
-L <i>[logfile]</i>	<code>LICD_LOGFILE [= file]</code>	Log information in a file (default: <code>licd.log</code>).
-l		Use the syslog (UNIX only).
-n		Run LicD, but not as a daemon (UNIX only).
-P <i>port</i>	<code>LICD_REM_SERVER_PORT = port</code>	Connect to port <i>port</i> on the remote license server (default: 8080). Possible ports you can use are 8080, 8936 or 80.
-p <i>port</i>	<code>LICD_LISTEN_PORT = port</code>	Listen on port <i>port</i> for communication with clients (default: 9090).
-Q <i>port</i>	<code>LICD_PROXY_PORT = port</code>	Specify the TCP port number to use for a proxy server (default: 3128).
-q <i>proxyserver[:port]</i>	<code>LICD_PROXY_SERVER = proxyserver[:port]</code>	If a proxy server is required for outgoing license requests, use this option to specify <i>proxyserver</i> as the HTTP proxy server. If you omit the <i>port</i> , the default port is 3128.

Option	Define in licdopt.txt	Description
-Rnum	LICD_AUTOREFRESH_TIMER = <i>num</i>	Auto-refresh timer. When the auto-refresh timer expires, the license server retrieves all licenses from either the license file (LICD_LICFILE) or from the remote license server. The number <i>num</i> specifies the refresh time in seconds. Valid values are between 60 and 14400 seconds (default: 600 seconds).
-r	LICD_USE_REMOTE	Use the remote license server to get client licenses in addition to a license file, or if no license file is present.
-v	LICD_LOG_VERBOSE	Verbose, repeat for more verbose.
	LICD_MASTER = <i>host[:port]</i>	In a redundant master - slave configuration, use this option to specify the master server in the option file on the slave server. See Section 3.4, Redundant Master - Slave Configuration .
	LICD_REFUSE_ANONYMOUS	Do not allow anonymous access to the local license server. This option forces that every client publishes his user name and system name with the client option TSK_NO_ANONYMOUS.
--install		Install the local license server as a Windows Service (Windows only).

Example of a local license server options file:

```
LICD_LICFILE = licdfile.txt
LICD_LOGFILE = mylog.log
LICD_ACCESSFILE = myveryspecialaccess.txt
LICD_LOG_VERBOSE
LICD_REM_SERVER=lic3.tasking.com:8080
LICD_LISTEN_PORT = 9090
LICD_PROXY_SERVER = proxy.mydomain.com
LICD_PROXY_PORT = 8081
LICD_AUTOREFRESH_TIMER = 300
LICD_USE_REMOTE
```

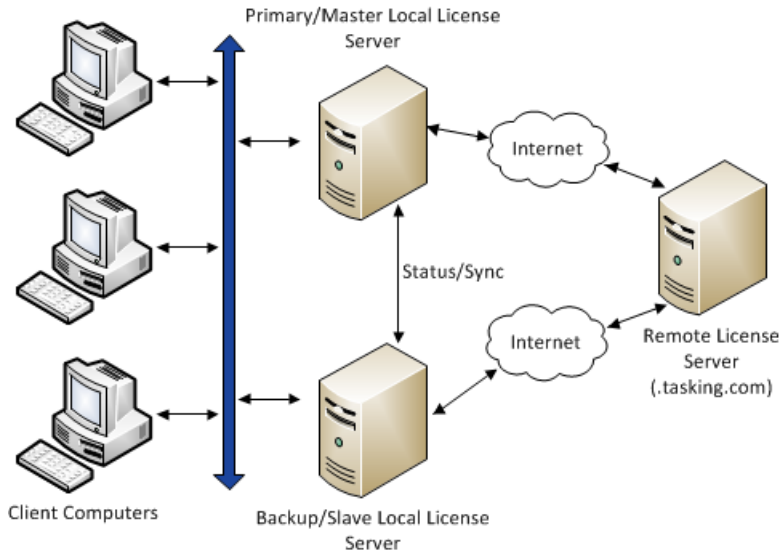
3.4. Redundant Master - Slave Configuration

In a redundant server configuration there is a master local license server and a slave local license server. In normal operation mode the master server serves the licenses for the clients and the 'Sleeping Slave' server synchronizes its own database every minute with the master database. The slave server refuses to serve licenses.

If the master becomes unreachable, the slave server changes its operation mode to 'Active Slave'. It tries to get a time limited slave license from a remote TASKING server and starts serving licenses with the last known master status. If the master server is back online again, the slave server changes its operation

mode to 'Sleeping Slave' and starts the sync again with its master. The status of the slave server is not synchronized back with the master.

A slave server cannot read license files and cannot get license data from a remote TASKING server except its own slave license. It can only get data from its master.



Client configuration

To use a redundant master - slave configuration, the client options file `licopt.txt` must contain both servers. For example,

```
TSK_LICENSE_SERVER = master.domain.com:9090;slave.domain.com:9090
```

If the master becomes unreachable a client switches to the next server in the list, the slave. If the master comes online again the slave refuses license requests. The client switches back to the first server in the list, the master.

A redundant local license server configuration needs two local license server licenses, one for the master and one for the slave (a time limited slave license).

The master and slave license servers must have the same version number and must run on the same operating system.

Slave server configuration

The file `licdopt.txt` of a slave server must contain the following configuration item:

```
LICD_MASTER=hostname[:port]
```

TASKING License Management

For example:

```
LICD_MASTER=lic-master.domain.com:9090
```

3.5. Creating a Local License Server License File

When you receive a license that must be serviced by a local license server then the email that you receive contains a license key and the license data. You have to copy the license data to the license file (default: licdfile.txt).

Example of a license email:

Hi user,

Thank you for ordering SW103800 (ARM toolset VX Package) version 5.1.

Product code: SW10380005044

The license key for your Floating server license for 1 seats is:

9ab1-46ba-8e0b-a86b

Serviced by the license server with Security ID: hgt1-KvZ4-Q70W-qBoG

Best regards,

The TASKING License Administrator

```
----- Copy the following data to your license file licdfile.txt -----
#
#
# TASKING License File
#
# Issue date: 2015-03-10 14:39:34
#
# License type: Floating server 1 seats license|Joinable
#
# Serviced by the license server with Security ID: hgt1-KvZ4-Q70W-qBoG
#
# License key: 9ab1-46ba-8e0b-a86b
#
# Product code: SW10380005044000
#
#
SW10380005044000=4EGRpf7IO0fj0e+tCdzw7QGjPFqnePFQK
MUTU62fXc0lCTx3rIAfRMYZ3+U0wsr5U631LvJ0pF8MNfoqd+l
```

zURBoKNTcn3d9By/a0eKxsxiZoOqaC23vvPmgqUzO3dluIgDKK
rSZhkI0w2M1YXzbFrqQ5fZv/oLOChTCORt9PfRY/YBxV+bmGyU
YKq+k++uBDOCbzVQsDrLefIuUNPipkEOEEHOFFre2HwUh4QKJn
zzDHw3+K4S4Bfn4CQVa6il1CflIOyu4d2JFUNUv4w==

TASKING License Management

Chapter 4. Administering Licenses

4.1. Monitoring the Status of the Local License Server

Browser

You can inspect the actual status and see the log information of the local license server with a browser. Use the following address:

```
http://lic_server_hostname:9090/stats
```

where, *lic_server_hostname* is the IP address or DNS name of the local license server.

The following columns are available:

Column	Description
License key	The license keys of the available licenses.
Product code	The product code.
Serial#	The serial number of the product.
Valid from	The date at which the license becomes valid.
Expires	The expiration date of the license, empty for perpetual licenses.
Modified	The date the license was modified.
Type	The type of license: Float (floating license), Node (node-locked license)
Seats	The number of seats for a floating license.
Seats in use	The number of licenses currently in use.

If you want to log the information in a file, you can use option **-L** of the local license server:

```
licd -L licd.log
```

Command line

Instead of using a browser you can also use the tool **licdstat** to see the status and log information of a local license server.

```
licdstat [option]...
```

The following options are available:

- afile** append output to a file, mutual exclusive with **-f**
- ffile** write output to a file, mutual exclusive with **-a**
- hhost** host to query (default: localhost)
- key** query a specific key

TASKING License Management

-l	list current license status
-pport	TCP port to use (default: 9090)
-t	display server status and version
-u	list current users
-v	verbose output

Note that the user name and system name in the users lists of a license, is only available if the option `TSK_NO_ANONYMOUS` is present in `licopt.txt`.

Example:

```
licdstat -l -u -t
```

```
LicD: vx.yrz localhost:9090 Tue 10 Mar 2015 09:24:54 CET SecurityID: 3cHE-GME0-FW3I-sHTW
```

```
08f3-0d7b-9fe9-25de SW16080002 v5.0 (floating ser#:99224421 seats:10 inuse:10)
```

Firstname1 Lastname1/pc_a	(v4.9 2015-03-10 09:24:11)
Firstname2 Lastname2/pc_b	(v5.0 2015-03-10 09:24:11)
Firstname3 Lastname3/pc_c	(v4.1 2015-03-10 09:24:12)
Firstname4 Lastname4/pc_d	(v4.6 2015-03-10 09:24:12)
Firstname5 Lastname5/pc_e	(v4.9 2015-03-10 09:24:13)
Firstname6 Lastname6/pc_f	(v4.4 2015-03-10 09:24:13)
Firstname7 Lastname7/pc_g	(v4.4 2015-03-10 09:24:14)
Firstname8 Lastname8/pc_h	(v4.4 2015-03-10 09:24:14)
Firstname9 Lastname9/pc_i	(v4.1 2015-03-10 09:24:15)
Firstname10 Lastname10/pc_j	(v5.0 2015-03-10 09:24:15)

```
5854-61b9-4ecb-04f0 SW10380002 v5.1 (floating ser#:99015724 seats:3 inuse:3)
```

Firstname11 Lastname11/pc_h	(v4.4 2015-03-10 09:24:16)
Firstname12 Lastname12/pc_i	(v4.1 2015-03-10 09:24:16)
Firstname13 Lastname13/pc_j	(v5.1 2015-03-10 09:24:17)

4.2. Transferring a License

It is possible to move a floating license to a node-locked client based license temporarily. This is called 'borrowing'. You can use the tool **licborrow** to obtain a time restricted node-locked client based license on a client system from a local TASKING server servicing a floating license for that client.

The license is retrieved by specifying the product code of the TASKING product. You can find the product code in the email or paper that contains the license key.

Note that you can only borrow a license for a client system that has already used that license before. Furthermore, you can restrict access to which users and systems are allowed to borrow a license. A 0 days request returns the borrowed license to the TASKING server.

To borrow a license (Windows)

1. From the **Start** menu, select **Programs » TASKING VX-toolset for target vx.yrz » Borrow a License**.

The Borrow a License dialog appears.

2. From the **Borrow license for** list, select the product for which you want borrow a license.
3. Enter the number of **Days** you want to borrow the license.
4. Click **Borrow a License**.

To borrow a license (UNIX)

On UNIX workstations, enter the following command at a shell command prompt, assuming that the `bin` directory of the TASKING product is in the search path:

```
licborrow [option]... product-code
```

The following options are available:

-ddays	number of days to borrow (0 .. 128), 0 returns a license
-v	verbose

Example

Borrow a license for 7 days:

```
licborrow -d7 SW10380005044
```

Return a license:

```
licborrow -d0 SW10380005044
```

4.3. Joining Licenses

You can bring together all licenses for one product into one overall license, this is called 'joining licenses'. Suppose you have placed two orders for a given product. In the first order you bought 5 floating licenses, in the second order you bought 10 additional floating licenses. With each order you have received a license key. You now have 15 floating licenses and 2 license keys. One key associated with 5 floating licenses, the other associated with 10 licenses.

If you run six instances of the product that use the license key obtained with the first order then one product instance will fail to execute due to the fact that there are only five licenses available. You do not want such behavior since you have bought 15 licenses. To solve this you can join licenses by updating the `licopt.txt` file by hand.

The following modification must be made in file `licopt.txt`: replace the license key by the string "PRODUCT".

TASKING License Management

For example:

```
# If you use a local license server can you can
# implement a bulk installation by using the
# Product code as license key.
TSK_LICENSE_KEY_SW103800 = PRODUCT
```

When the product validates its license the license server resolves the license request by scanning through all of its licenses for the given product.

4.4. Bulk Installation

If you have to install a product on a large number of computers you may want to distribute one and the same instance of a TASKING product including the license files (`licopt.txt`) to all computers involved without having to execute the install program on each computer. This functionality is called "bulk installation".

To facilitate a bulk installation

1. Install a TASKING product in the regular way as described in [Chapter 2, Licensing TASKING Products](#).
2. Join the licenses. Modify file `licopt.txt` as follows:

Replace the license key by the string "PRODUCT". For example:

```
# If you use a local license server can you can
# implement a bulk installation by using the
# Product code as license key.
TSK_LICENSE_KEY_SW103800 = PRODUCT
```

3. Copy the installed product tree, including file `licopt.txt`, to all computers.

When the product validates its license the license server resolves the license request by scanning through all of its licenses for the given product.

4.5. Restricting Access to Licenses

With an access file (default: `licdaccess.txt`) you can deny, permit, block and bind users and systems to licenses. Without an access file no restrictions apply, every user, system or IP address is allowed access. See option **-A** in [Section 3.3, Local License Server Options](#).

By default, license requests are anonymous. To see which user name and system name are sent with a license request, you need to add the option `TSK_NO_ANONYMOUS` to your local `licopt.txt` file. See [Section 2.4.1, Licensing Options File](#).

The commands in an access file are processed from top to bottom. A subsequent command can overrule a previous command.

After a change in the access file the license server must be restarted.

The following commands are possible in an access file.

deny

```
deny user { all | name }
deny system { all | name }
deny ip { all | ipaddr[-ipaddr] }
```

With this command the local license server denies access to the local license server for a user, a system, an IP address or a range of IP addresses.

permit

```
permit user { all | name }
permit system { all | name }
permit ip { all | ipaddr[-ipaddr] }
```

With this command the local license server permits access to the local license server for a user, a system, an IP address or a range of IP addresses.

block

```
block user { all | name } to key
block system { all | name } to key
block ip { all | ipaddr[-ipaddr] } to key
```

With this command the local license server blocks access to a specific license for a user, a system, an IP address or a range of IP addresses.

bind

```
bind user { all | name } to key
bind system { all | name } to key
bind ip { all | ipaddr[-ipaddr] } to key
```

With this command the local license server allows access to a one specific license for a user, a system, an IP address or a range of IP addresses. A subsequent `bind` command can overrule a previous `bind` command.

borrow deny

```
borrow deny user { all | name } from { all | key }
borrow deny system { all | name } from { all | key }
borrow deny ip { all | ipaddr[-ipaddr] } from { all | key }
```

With this command the local license server denies a user, a system, an IP address or a range of IP addresses to borrow any license or a specific license.

borrow permit

```
borrow permit user    { all | name } from { all | key } days n
borrow permit system { all | name } from { all | key } days n
borrow permit ip      { all | ipaddr[-ipaddr] } from { all | key } days n
```

With this command the local license server permits a user, a system, an IP address or a range of IP addresses to borrow any license or a specific license for a maximum number of days.

Examples

To allow access to user `eric` only, enter:

```
deny user all
permit user eric
```

To specify that system `svr100a` is not allowed to access license key `aaaa-bbbb-cccc-dddd`, enter:

```
block system svr100a to aaaa-bbbb-cccc-dddd
```

To specify that all users are bound to license key `xxxx-xxxx-yyyy-yyyy`, except for users `hans` and `martin` and that they are bound to license key `1234-5678-9012-3456` and `aaaa-bbbb-cccc-dddd` respectively, enter:

```
bind user all to xxxx-xxxx-yyyy-yyyy
bind user hans to 1234-5678-9012-3456
bind user martin to aaaa-bbbb-cccc-dddd
```

To bind all users to the (product code) license `3de8-76aa-ed65-3271`, except for two development departments:

```
bind ip all to 3de8-76aa-ed65-3271
bind ip 192.168.1.0-192.168.1.100 to 6a3c-cc42-1b40-a4ec
bind ip 192.168.1.101-192.168.1.200 to 11fa-5eca-7fee-1cd8
```

To allow an IP address range to borrow a license from `6a3c-cc42-1b40-a4ec` for a maximum of three days, enter:

```
borrow deny ip all from all
borrow permit ip 192.168.1.0-192.168.1.100 from 6a3c-cc42-1b40-a4ec days 3
```