LICMAN User's Manual
Version 2.0.20

·· T ·· ·· System s·· ·· ··

Imprint

Publisher

T-Systems International GmbH
System Integration

D-70771 Leinfelden-Echterdingen

Filename	Document Number	Document Designation
LICMAN	LICMAN-2.0.20-MANUAL-13/01-e	User's Manual

Version	Date of Document
2.0.20	January 2013

Author

PLM Applications

Contact	Phone / Fax	E-Mail / Internet
CADCAM Hotline	+49-711-34243-2603	cadcam.hotline@t-systems.com
		https://servicenet.t-systems.com/plm-applications

Short Description

LICMAN User's Manual

Table of Contents

1.   General Remarks

1.1.   Purpose

This manual describes the installation and functions of the T-Systems license manager product LICMAN.

1.2.   Installation Procedure

The installation includes the copying of data from installation medium, and a few other steps. Perform the installation by the following steps:

1. Familiarize yourself with the prerequisites (chapter 2)

2. Prepare the installation (chapter 3)

3. Perform the installation (chapter 4)

4. Start License Manager LICMAN (chapter 5)

1.3.   Conventions

`Bold Courier` font indicates program, file names, environment variable names, registry entries and commands to enter.
Variable values are embraced in pointed brackets (**`<variable>`**).

1.4.   Windows Registry Considerations

LICMAN uses Windows registry to store a few information used by LICMAN components.

On 32 bit Windows operating systems this information is stored in registry node
`HKEY_LOCAL_MACHINE\SOFTWARE\T-Systems\Licman`

On 64 bit Windows operating systems this information is stored in registry node
`HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\T-Systems\Licman`

A typical German Windows registry entry gets displayed by registry editor like this:
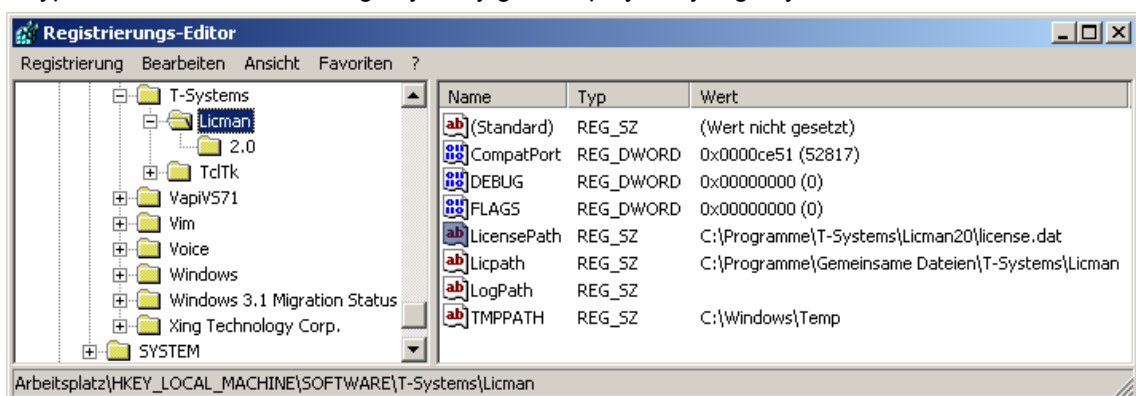
Note:

This document generally uses registry entry names for 32 bit Windows operating systems. Wherever this document mentions registry node
`HKEY_LOCAL_MACHINE\SOFTWARE\T-Systems\Licman,`
on 64 bit Windows operating systems registry node

`HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\T-Systems\Licman`
has to be used instead.

1.5.   Time Considerations

Time used by LICMAN is UTC (universal time, coordinated). In LICMAN context, UTC is the same as Greenwich Mean Time.

Date and time specified in license file is UTC, as well as time reported in output files.

To query current UTC time, use UNIX command `date -u.`

2. Prerequisites

2.1. Operating System Versions

LICMAN is currently available for following platforms:

Vendor	Operating system	Processor architecture
IBM	AIX 5.1 [1]	32/64 bit
HP	HP-UX 11.00 (PA-RISC Platform) [2]	32/64 bit
SGI	IRIX 6.5	32/64 bit
SUN	SunOS 5.8 (SPARC Platform) [3]	32/64 bit
Linux	Kernel 2.6.9 (x86 Platform) [4]	32 bit
	Kernel 2.6.9 (x86-64 Platform) [5]	64 bit
Microsoft	Windows (x86 Platform) [6]	32 bit
	Windows (x86-64 Platform) [7]	64 bit

[1] Usable for AIX 5.1 and newer

[2] Usable for HP-UX 11.00 and newer

[3] Usable for SunOS 5.8 and newer

[4] Usable for kernel 2.4.21 (e.g. SuSE Linux 9.0, Red Hat Enterprise Linux 3) and newer

[5] Usable for kernel 2.4.21 (e.g. SuSE Linux 9.0 for x86-64, Red Hat Enterprise Linux 3 for x86-64) and newer

[6] Usable for Windows 2000 and newer

[7] Usable for Windows XP Professional x64 Edition and newer

2.2. Disk Requirements

1 MB of disk space for each installed operating system platform will be sufficient.

3. Preparing the Installation

LICMAN is part of several T-Systems product distributions. The installation scripts of those products may install LICMAN as part of the product's installation process or may ask during the product's installation whether or not to install LICMAN.

No separate LICMAN installation medium may be supplied nor required. In this case, skip chapter 3.1 "Define the Installation Directory" and chapter 3.2 "Read the LICMAN installation medium".

3.1. Define the Installation Directory

LICMAN installation medium may be copied into any directory, if sufficient free disk space is available in that file system. The installation directory is queried during the installation.

3.2. Read the LICMAN installation medium

UNIX:

The LICMAN installation medium is written with the command `tar` and contains the following files:

`./tmplic/inst`	installation script
`./tmplic/<os>/bin/licman20_gld`	GLD (Global License Daemon) executable
`./tmplic/<os>/bin/licman20_lld`	LLD (Local License Daemon) executable
`./tmplic/<os>/bin/licman20_appl`	testing utility for GLD and LLD
`./tmplic/<os>/bin/licman20_id`	LICMAN id executable
`./tmplic/bin/licman20`	script for starting/stopping license daemons
`./tmplic/doc/licman20.pdf`	this manual (PDF document)

where `<os>` stands for the corresponding operation systems.

To read the data the following steps are to be carried out:

1. Change to any temporary directory with enough disk space:
   `cd <temporary_directory>`
2. Read the data from the installation medium by command:
   `tar -xvf <installation_medium_name>`

The temporary directory now contains following files and directories:

`./tmplic/bin`	ksh scripts
`./tmplic/<os>/bin`	executables
`./tmplic/inst`	ksh installation script
`./tmplic/doc`	this manual

Windows:

The LICMAN installation medium contains the installation package files for Windows and the installation startup program `setup.exe`.

3.3. Stop License Manager

If you plan to replace LICMAN Version 1.2 by LICMAN Version 2.0 make sure that the LICMAN Version 1.2 LLD license process and possibly GLD license process were stopped.

UNIX:

Use the `kill` command for stopping the old license daemons `licman12_lld` and possibly `licman12_gld`. Root permissions may be required to stop running processes.

Windows:

Login as administrator for stopping the service `Licman LLD` (and possibly `Licman GLD`).

If you plan to replace existing LICMAN Version 2.0 executables with new ones make sure that the license daemons LLD and possibly GLD were stopped.

UNIX:

Use the `kill` command for stopping the old license daemons `licman20_lld` and possibly `licman20_gld`. Root permissions may be required to stop running processes.

Windows:

Login as administrator for stopping the service `Licman 2 LLD` (and possibly `Licman 2 GLD`).

Stopping and/or disabling a running Windows service is performed by administrative tool "Services" found by following "Start -> Settings -> Control Panel -> Administrative Tools":

T-Systems International GmbH, January 2013
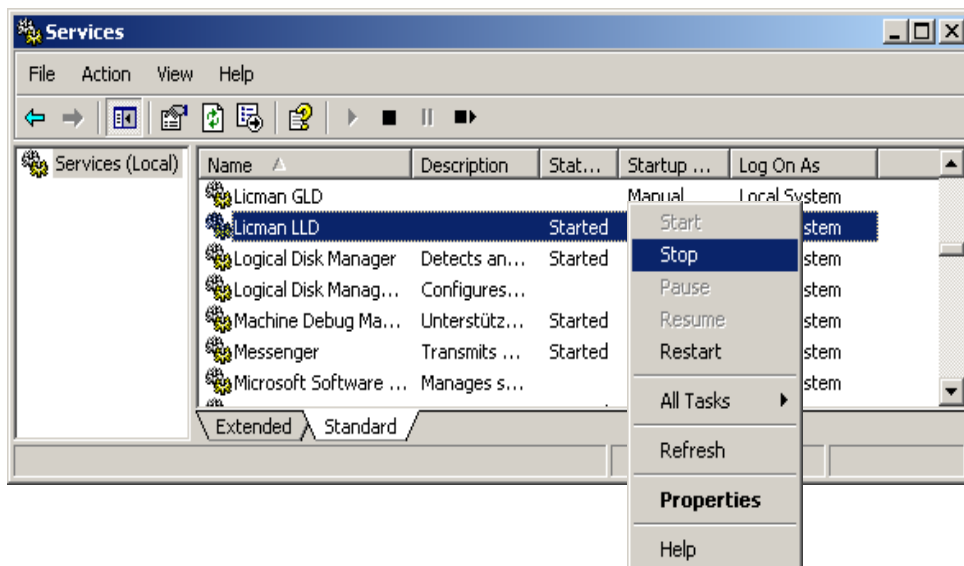
3.4.   Disable/uninstall previous LICMAN 1.2

As LICMAN 2.0 serves as license manager for LICMAN 1.2 applications too, LICMAN 1.2 and LICMAN 2.0 do not run concurrently.

UNIX:

Prevent LICMAN 1.2 daemons to get restarted.

Windows:

Disable LICMAN 1.2 services before installing LICMAN 2.0. The startup mode of LICMAN 1.2 services has to be set to "Disabled" or "Manual" in order to prevent a restart at reboot. If desired, LICMAN 1.2 may be uninstalled *before* installation of LICMAN 2.0. Uninstalling LICMAN 1.2 software on Windows PCs *after* LICMAN 2.0 installation may require re-installation of LICMAN 2.0, because some common registry entries are removed by uninstallation of LICMAN 1.2.

Disabling a Windows service is performed by administrative tool "Services" found by following "Start -> Settings -> Control Panel -> Administrative Tools" and editing the service's properties.

Alternatively you may disable LICMAN services by running "Start -> Settings -> Control Panel -> Software". Select "T-Systems Licman 1.2", choose "Change/Remove". Choose "Modify" in the appearing dialog box and modify services startup modes to "disabled".

4.　　Performing the Installation

Performing a LICMAN installation requires the following steps:

1. Execute the installation startup script setup
   (refer to chapter 4.1 "Running the UNIX Installation Startup Script "inst""
   and to chapter 4.2 "Running the Windows Installation Startup Program "setup.exe"")

2. Get the LICMAN System Id
   (refer to chapter 4.3 "Getting the LICMAN System Id")

3. Get a license from T-Systems delivery department

4. Make the license available
   (refer to chapter 4.4 "Making a LICMAN License available")

5. Verify the installation:
   (refer to chapter 5.4 "Verifying the LICMAN Installation")

6. Start LICMAN
   (refer to chapter 5 "Starting LICMAN")

Remark:
LICMAN is part of several T-Systems product distributions. The installation scripts of those products may include installing LICMAN. In this case, skip chapter 4.1 "Running the UNIX Installation Startup Script "inst"" and 4.2 "Running the Windows Installation Startup Program "setup.exe".

4.1.　　Running the UNIX Installation Startup Script "inst"

Skip this chapter if LICMAN installation has been installed as part of another T-Systems product installation.

The installation of LICMAN is performed by the install script **inst**. This install script puts the files of LICMAN to be installed into the desired directory.

Invoke **inst** with the command **<temporary_directory>/tmplic/inst**

The installation process is interactive and occurs in the following steps:

1. Specify operating system:
   Modules are available for the following operating systems:
   <os> <os> <os> … <os>
   Specify operating system identifier [<current_operating_system>] ?

2. Specify installation directory:
   Specify installation directory [<current_directory>] ?

3. Installing the software
   Using installation directory <installation_directory> ...
   or
   Creating installation directory <installation_directory> ...
   Installing shell scripts ...
   Installing modules for operating system <operating_system> ...
   Installing documents ...

4. Remove the installation directory (optional)
   Remove temporary installation directory tmplic (y/n) [n] ? n

   Keeping installation directory tmplic ...

   Installation finished.

Note:
Each call of `inst` LICMAN does install one operating system. You should not delete the installation directory `tmplic` until LICMAN has been installed for all operating systems needed.

4.2. Running the Windows Installation Startup Program "setup.exe"

Skip this chapter if LICMAN installation has been installed as part of another product installation.

4.2.1. Interactive Installation

Interactive installation of LICMAN for Windows is performed by the startup program `setup.exe`. To run `setup.exe`, login as administrator and double-click the file `setup.exe` in setup directory of installation medium.
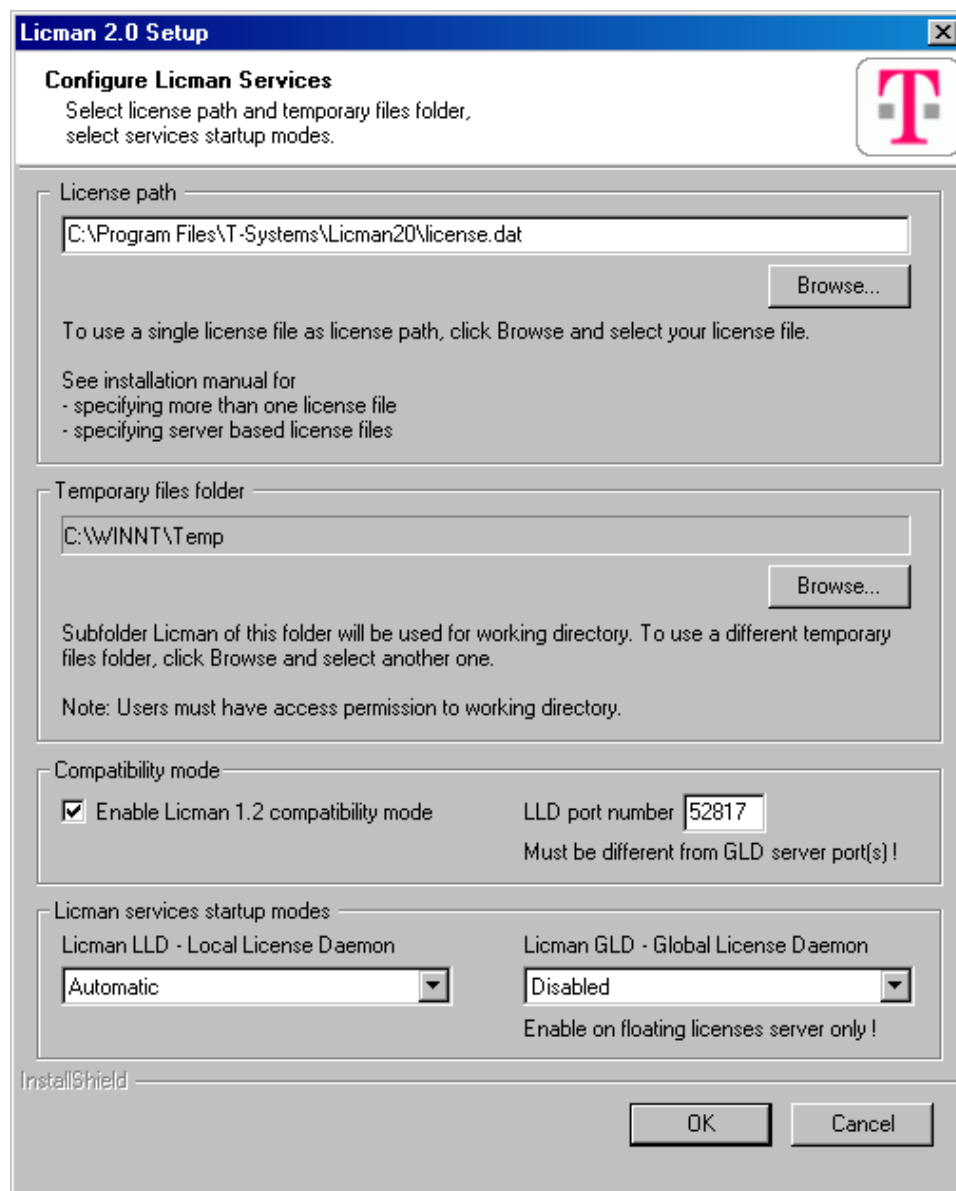
During the installation of LICMAN, you have to specify the license path. If the license path consists of one single license file, you may type the license file's name or you may use "Browse" to locate an already existing license file. How to provide license file(s) see chapter 4.4 "Making a LICMAN License available".

The temporary files folder is used to determine LICMAN's working directory. For more information about working directory see chapter 6.2.3 "Working Directory".

LICMAN compatibility mode is required to run applications developed using LICMAN 1.2. Usually application versions older than January 1st, 2008 are concerned.

The LICMAN setup installs the LLD (Local Licenses Daemon) and the GLD (Global License Daemon). LLD and possibly GLD will run as Windows services. During LICMAN setup you have to specify the startup mode of LLD and GLD. The standard setting - if workstation is no license server - should be "Automatic" for LLD and "Disabled" for GLD. If the workstation is a network i.e. floating licenses server, use settings "Automatic" for LLD and GLD.

Re-running LICMAN setup after installation allows reconfiguring, repairing or removing an existing installation.
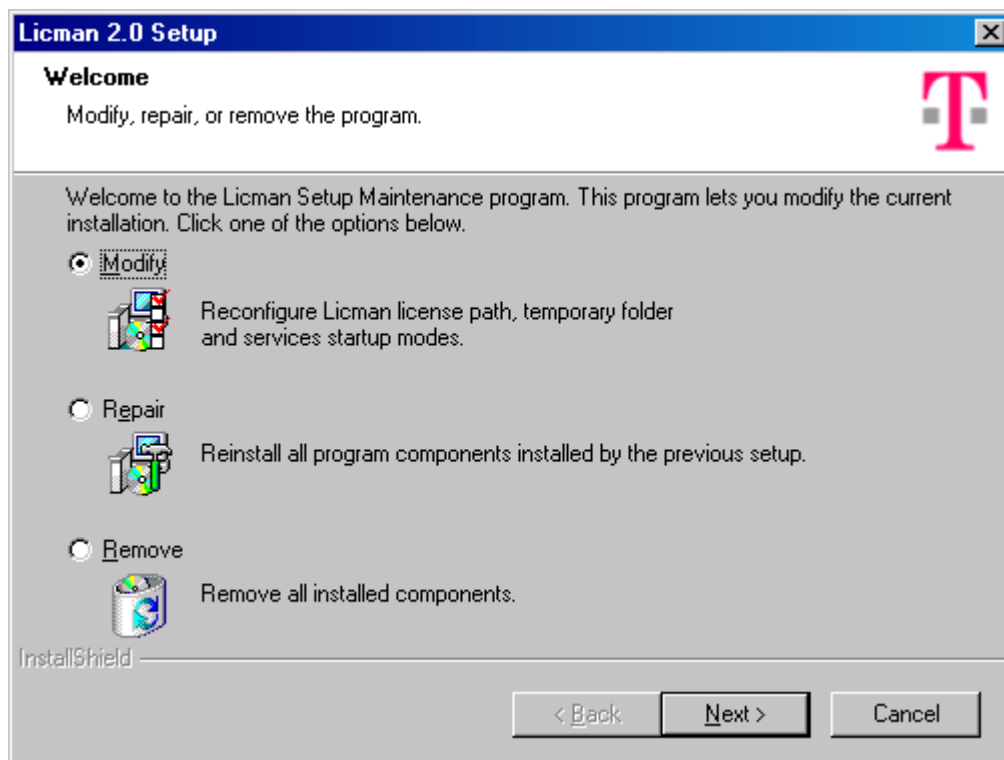
4.2.2. Silent Installation

Silent installation of LICMAN requires an existing **InstallShield Silent Response File** (.iss file).

Silent installation procedure:

1. Login as administrator.
2. Create a response file e.g. named `C:/temp/licman.iss` running `setup.exe` from console window using syntax **setup.exe -f1C:/temp/setup.iss –r** and follow graphical user interface. Please note, LICMAN does not get installed but only response file gets created.

3. Modify response file according to your needs. All values set during graphical user interface are easily to be located and changeable.

4. Run silent installation on an arbitrary PC as administrator using syntax
   `setup.exe -f1C:/temp/setup.iss -s`.

For convenience, a template response file `setup.iss` is provided in installation directory.

Note:
Name of response file has always to be specified with full path name.

4.3. Getting the LICMAN System Id

You get the LICMAN system id(s) by running the LICMAN id utility (see chapter 7.2 "LICMAN Id Utility" below).

The LICMAN system id is required by the T-Systems delivery department for creating a valid license file for your desired Workstation.

Remarks:
LICMAN system ids are not case sensitive. Id strings are valid whether they are uppercase, lowercase or mixed case characters.

On

- a Windows PC with several network adapter cards
- a Linux PC with several network adapter cards
- AIX 5.1 or newer

you may get more than one possible LICMAN system id. Each of them is accepted by LICMAN as a valid system id.

You may use following alternate command on UNIX workstations to get a valid LICMAN id:

Vendor	Operating sytem	Command
Hewlett-Packard	HPUX	/usr/bin/uname -i
IBM	AIX 5.2 and older AIX 5.3 or newer, AIX 5.1/5.2 with PTFs	/usr/bin/uname -m /usr/bin/uname -F
SiliconGraphics	IRIX	/sbin/sysinfo -s
SUN Microsystems	SunOS	/usr/bin/hostid

4.4. Making a LICMAN License available

UNIX

In order to successfully start a LICMAN daemon, a license path has to be defined by environment variable `LICMAN_LICENSE_PATH` or alternative environment variables (see chapter 6.2.1 "License Path" below).

To make a license available set environment variable(s) as described in chapter 6.2.1 "License Path" below before starting LICMAN.

If no license path is defined, some T-Systems products use a default license path. The T-Systems product MEDINA for example uses the default license path
`<installation_directory>/cae/prod/data/licman/license.dat`
where `<installation_directory>` is the product's installation directory.

Windows

In order to successfully start a LICMAN service, a license path has to be defined by registry key **LicensePath** or alternative registry keys (see chapter 6.2.1 "License Path" below). This registry key will be created by LICMAN installation and will have a value as specified at installation time.

At installation time, a license path **<installation directory>\license.dat** gets proposed. If you did not change the proposed license path

- either provide a license file <installation directory>\license.dat

- or adjust registry key(s) to your needs as described in chapter 6.2.1 "License Path" below.

If not sure about license path defined at LICMAN installation time, use Windows registry editor and locate registry key **HKEY_LOCAL_MACHINE\SOFTWARE\T-Systems\Licman**. Entries of key show currently set values (see chapter 1.4 "Windows Registry Considerations" above)

Alternatively license path may be specified UNIX-like by environment variable (see above). If both environment variable and registry key are set, environment variable takes priority over registry key.

5.   Starting LICMAN

Important note:

Running LICMAN executables uses a working directory for creating, reading and writing temporary files (see chapter 6.2.3 "Working Directory" below).

These files are required for communication between LICMAN components: UNIX sockets, etc... Even if LICMAN will try to create new ones, if these files are deleted while LICMAN is running, errors may or may not occur.

Make sure to prevent, that working directory and temporary files do not get deleted while LICMAN executables are running.

Execution of LICMAN requires additional information to be set. This information is read from environment variables on UNIX and from registry entries and/or environment variables on Windows. Environment variables and registry entries are described later.

5.1.   Starting LICMAN on UNIX

5.1.1.   Starting Global License Daemon (GLD)

Only one single LICMAN GLD process is able to run per workstation. An attempt to start an additional GLD will detect running GLD and will terminate without raising an error condition. The GLD is started by calling executable

`<installation_directory>/licman20_gld`

or better by calling startup script

`<installation_directory>/licman20 -start gld`

The second form requires environment variable `LICMAN_INSTALL_PATH` to be set before, if script `licman20` and executable `licman20_gld` are not in same directory.

In case of an error, an error message is written into the message log file (see chapter 6.2.5 "Message Log File").

For floating i.e. network licenses, a licensed host is required to have started GLD successfully. Node locked licenses do not require a GLD to be started. If no GLD is running for network licenses, no floating license will be available.

5.1.2.   Starting Local License Daemon (LLD)

Only one single LICMAN LLD process is able to run per workstation. An attempt to start an additional LLD will detect running LLD and will terminate without raising an error condition. The LLD is started by calling

`<installation_directory>/licman20_lld`

or better by calling startup script

`<installation_directory>/licman20 –start lld`

The second form requires environment variable `LICMAN_INSTALL_PATH` to be set before, if script `licman20` and executable `licman20_lld` are not in same directory. Starting LLD by script is preparing LICMAN 1.2/LICMAN 2.0 compatibility mode by creating a compatibility license file `licenses` in working directory `/tmp/.licman`. To enable compatibility mode, set environment variable `debisLICDIR` to `/tmp/.licman` *before* calling startup script `licman20` and *before* running applications developed using LICMAN 1.2. Application versions older than January 1st, 2008 usually are concerned.

In case of an error, an error message is written into the message log file (see chapter 6.2.5 "Message Log File").

For node locked and floating licenses, application's host is required to start a LLD successfully. If no LLD is running, no license will be available.
If no LLD is running, some T-Systems product will start a LLD by itself. This product expects LICMAN to be installed in a product specific default directory. If your LICMAN installation uses another directory, you should specify your LICMAN installation directory by environment variable `LICMAN_INSTALL_PATH` for this product and consult product's installation manual in this case.

5.2. Starting LICMAN on Windows

5.2.1. Starting Global License Daemon (GLD)

5.2.1.1. Starting Global License Daemon as Service

Only one single LICMAN GLD process is able to run per workstation. An attempt to start an additional GLD will detect running GLD and will terminate. If LICMAN GLD process is already running, a GLD additionally started in debug mode will terminate without raising an error condition. If LICMAN GLD process is already running, an additionally started GLD service will terminate and Service Control Manager will write startup failure event information into system event log.

After restarting the computer, the service `Licman 2 GLD` will be started from operating system if automatic startup mode is enabled. Login as administrator to stop or restart this service or to change startup mode.

In case of an error, an error message is written into the message log file (see chapter 6.2.5 "Message Log File").

For floating i.e. network licenses, a licensed host is required to have started GLD successfully. Node locked licenses do not require a GLD to be started. If no GLD is running for network licenses, no floating license will be available.

5.2.2. Starting Local License Daemon (LLD)

Only one single LICMAN LLD process is able to run per workstation. An attempt to start an additional LLD will detect running LLD and will terminate. If LICMAN LLD process is already running, an LLD additionally started as detached process or started in debug mode will terminate without raising an error condition. If LICMAN LLD process is already running, an additionally started LLD service will terminate and Service Control Manager will write startup failure event information into system event log.

5.2.2.1. Starting Local License Daemon as Service

After restarting the computer, the service `Licman 2 LLD` will be started from operating system if automatic startup mode is enabled. Login as administrator to stop or restart this service or to change startup mode.

5.2.2.2. Starting Local License Daemon as detached Process

As starting and stopping Windows services requires special permissions, an additional startup mode has been made available with LICMAN version 2.0.9. Normal users are now allowed to startup `Licman 2 LLD` as "detached" process.

A "detached" process is a console process which does not inherit its parent's console window. Because windows console processes started by command line from console or batch script usually inherit its parent's console and get killed from operating system if console window gets closed, LICMAN LLD process in case of an inherited console window restarts running as a new "detached" process and finishes current "attached" process, i.e. process with console.

Run LICMAN LLD

- in console window by command **`<installation_directory>/licman20_lld.exe`**

- in batch file by command **`<installation_directory>/licman20_lld.exe`**

- by double clicking **`licman20_lld.exe`** in Windows explorer

- by creating a scheduled task starting LICMAN LLD at Windows start

- et cetera

Please note:

All processes started during session after logon get killed from Windows operating system at end of session after logout. Only processes started as service by Windows Service Control Manager or as scheduled task by Windows Task Scheduler are able to survive login/logout of users.

If a LICMAN licensed application does not get finished ordinarily but by simply logging out from Windows session and LICMAN LLD process has been started other than by Service Control Manager or by Task Scheduler, LICMAN LLD process gets killed by Windows operating system at logout along with application and is therefore no longer able to free application's license. In case of a floating license, this license will keep allocated until next restart of LICMAN LLD on same PC.

5.3.   Starting LICMAN in debug mode

In order to find the reason of unknown LICMAN problems, it may be helpful to run LICMAN GLD and/or LLD in debug mode. When started in debug mode, LICMAN GLD and/or LLD write debug output to console window.

The debug level controls the amount of debug output.

Debug level	Description
0	No debug output will be produced
1	Debug output without reporting communication details between LICMAN components will be produced
2	Debug output including communication details between LICMAN components will be produced

Debug output may be redirected to a file instead of console output. Use command syntax **`command > file`** to redirect standard output of command to file.

Using UNIX Debug Mode

On UNIX operating systems, a starting GLD and/or LLD is running in debug mode if environment variable **`LICMAN_DEBUG`** exists and has a value greater than 0. Debug mode remain active while GLD and/or LLD daemon is running. Debug output level may change by OPTION entry found in license file.

Enable Debug Mode

- Stop running GLD and/or LLD daemon

- Open console window
- Set environment variable **`LICMAN_DEBUG`** to a value greater than 0
- Run GLD and/or LLD executable. Executable will run in debug mode.

Disable Debug Mode

- Stop running GLD and/or LLD executable.
- Unset environment variable **`LICMAN_DEBUG`**
- Restart GLD and/or LLD. Executable will run as daemon.

Using Windows Debug Mode

On Windows operating systems, a starting GLD and/or LLD is running in debug mode if registry key

HKEY_LOCAL_MACHINE\SOFTWARE\T-Systems\Licman\Debug

exists and has a value greater than 0.

Alternatively debug mode may be specified UNIX-like by environment variable (see above). If both environment variable and registry key are set, environment variable takes priority over registry key.

Execution flags remain active while GLD and/or LLD daemon is running. Debug output level may change by OPTION entry found in license file.

Registry import files with suffix **`.reg`** are supplied in LICMAN installation directory to switch debug mode on and off. Double click file **`DebugOn.reg`** (**`DebugOn64.reg`** on 64 bit windows) to enable debug mode and **`DebugOff.reg`** (**`DebugOff64.reg`** on 64 bit windows) to disable debug mode

Enable Debug Mode

- Stop running GLD and/or LLD service.
- Set registry key to value greater than 0
- Open console window
- Run GLD and/or LLD executable by pathname of executable. Executable will run in debug mode.

Disable Debug Mode

- Stop running GLD and/or LLD executable.
- Set registry key to value 0
- Restart GLD and/or LLD service.

5.4.   Verifying the LICMAN Installation

You can verify a successful LICMAN installation by executing the LICMAN test utility as described below (see chapter 7.1 "LICMAN Test Utility")

6. Description of LICMAN

Following T-Systems products are controlled by the license manager LICMAN:

- MEDINA
- VAMOS
- COM-STEP, COM-VDAFS
- NIS-CAD
- CATIA IUA macros
- CMI, CMI-RII
- PDM-Webconnector, PDM-Workbench
- ...

In a heterogeneous workstation network, LICMAN maintains two different license types:

- Network licenses
- Node locked licenses

6.1. License Daemons
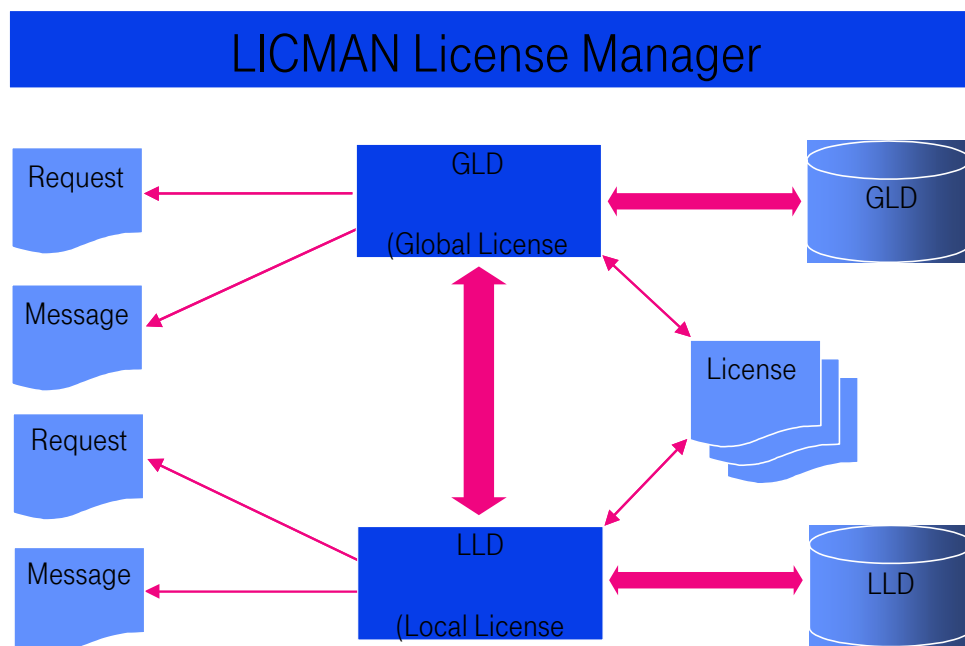
LICMAN consists of the following two components:

- "Global License Daemon" (GLD)
- "Local License Daemon" (LLD)

For network licenses, both components are required. For node locked licenses, no GLD is needed.
Licenses are requested by the application via LLD. In case of a node locked license, the LLD

performs the allocation of licenses. In the network case, the request is transferred to the GLD(s).

Floating licenses with more than one LICMAN server id require a license server system of **redundant license servers**. A GLD daemon on each server is needed. Redundant license servers together manage the same set of licenses. A license allocation request is granted, if more than half of servers allow allocation and no server explicitly denies allocation. If for example 1 of 3 redundant servers fail, to servers may still allow allocation of a license.

6.2.   License Manager Files

LICMAN uses following types of files:

- license files as parts of license path
- working directory
- log file directory
- message log file
- request log files (optional)
- temporary files in working directory

All files and directories must have read/write-access for GLD and/or LLD, except license files. License files must have read-access for GLD and/or LLD

6.2.1.   License Path

LICMAN expects its license files to be found by a license path.

A license path is an enumeration of license sources. Licenses sources are separated by a separator character. On UNIX operating systems, the separator character is a colon. On Windows operating systems, the separator character is a semicolon.

Each item of this enumeration is

- either the fully qualified pathname of a locally accessible license file.
  Each license file contains SERVER, LICENSE and/or OPTION entries (see below).
- or is a server based license source of type **Port@Server**
  The license information is supplied by LICMAN GLD on given server waiting for incoming request on given port.

Several license files and/or server based license sources may be combined to one license path.

A license path of type **Port1@Server1,Port2@Server2,Port3@Server3** is usually used for a license server system consisting of **3 redundant servers**.

Licenses are managed by following rule:

- Licenses are searched in license sources as specified. License source 1 is searched 1st, license source 2 is searched 2nd, …
- Licenses within one license source are searched top down. If a license source contains a node locked license 1st and a floating license for same product 2nd, the node locked license is tried 1st.

License paths are checked periodically, every 60 seconds by GLD and every 15 seconds by LLD. If the license path definition is found to have changed or if contents of a locally accessible license file are found to have changed, changes are accepted by LICMAN. Thus, changes in license files will get active within 1 minute without restarting LICMAN GLD and/or LLD.

UNIX License Path

The license path is defined by environment variable `LICMAN_LICENSE_PATH.`

If GLD and LLD require different license paths, environment variables `LICMAN_LICENSE_PATH_GLD` and/or `LICMAN_LICENSE_PATH_LLD` may be used instead.

If both environment variables `LICMAN_LICENSE_PATH_GLD` and `LICMAN_LICENSE_PATH` are set, LICMAN GLD uses `LICMAN_LICENSE_PATH_GLD.`

If both environment variables `LICMAN_LICENSE_PATH_LLD` and `LICMAN_LICENSE_PATH` are set, LICMAN LLD uses `LICMAN_LICENSE_PATH_LLD.`

License path example:

Source1:Source2:Source3:...:SourceN

Windows License Path

The license path is defined by registry key
`HKEY_LOCAL_MACHINE\SOFTWARE\T-Systems\Licman\LicensePath`

If GLD and LLD require different license paths, registry keys
`HKEY_LOCAL_MACHINE\SOFTWARE\T-Systems\Licman\LicensePathGLD`
and
`HKEY_LOCAL_MACHINE\SOFTWARE\T-Systems\Licman\LicensePathLLD`
may be used instead.

If both registry keys `LicensePathGLD` and LicensePath are set, LICMAN GLD uses `LicensePathGLD`.

If both registry keys `LicensePathLLD` and LicensePath are set, LICMAN LLD uses `LicensePathLLD`.

Alternatively license path may be specified UNIX-like by environment variable (see above). If both environment variable and registry key are set, environment variable takes priority over registry key.

License path example:

Source1;Source2;Source3;...;SourceN

Updating current UNIX License Path

On UNIX operating systems, a starting GLD and LLD daemon saves current license path information from environment to LICMAN working directory. Running GLD and LLD daemons check license path information in LICMAN working directory.

To notify a running LICMAN daemon about change of license path, set environment variable value to new license path and start an additional GLD and/or LLD executable. This additional executable will update license path information in LICMAN working directory and exit afterwards, because an already running GLD and/or LLD executable is detected.

Updating current Windows License Path

On Windows operating systems, GLD and LLD services always check current license path information from registry key.

To notify a running LICMAN service about change of license path, change value of registry key.

If license path is alternatively specified by environment variable, a restart of GLD and/or LLD service is required for new license path to take effect.

6.2.2.   License File

Each license file may contain

- Floating and/or node locked license entries
- Server definition entries
- Execution option entries
- Comment lines
- Empty lines with white spaces

License files are subject to following rules:

- Any leading and/or trailing space/tab characters in a line are ignored
- Multiple contiguous space/tab characters in a line are treated as one single space character
- Carriage return characters (hex '0D') at the end of a line are ignored, i.e. UNIX and Windows text files are equivalent
- Empty lines are ignored
- Continued lines ending with a \ character are treated as one single line
- Comment lines starting with a # character are ignored
- Trailing comments starting with a # character are ignored
- Entries after concatenating continued lines have following syntax
  Keyword Parameter[=Value] [Parameter[=Value] ...]
  Each parameter must not appear more than once.
  Parameter names are not case sensitive.
  In case of multiple definitions of a parameter, the last one gains.
  String values containing special characters like spaces etc. are to be enclosed by apostrophes, e.g. PARAM="String value"

6.2.2.1.   Server definition entries

A server definition entry assigns a physical license server to a floating license's LICMAN system id.

Notes:

- SERVER entries have to be supplied by customer for each license server.
- SERVER entries contained in a network license source are inherited by LLD from GLD.
- The customer may receive a license file containing one or more SERVER entry templates, where HOSTID(s) and PORT values are proposed. The proposed PORT value is usually the same as LICMAN 1.2 GLD port. Templates have to be modified according customer's need and uncommented. E.g. HOSTNAME templates have to be replaced by real host names.
- TCP port number PORT may modified by customer if conflicting with a port number already used by another product/service. Take care not to use a port number used by any other running process.
- Each GLD daemon is listening on 1 dedicated port for incoming LLD requests. LLD daemons are using this port to connect to GLD. Thus all license files have to use same port number for same license server.
- Linux or Windows PCs may have more than 1 network adapter and therefore more than 1 valid LICMAN server id. All LICMAN server ids of a host used as HOSTIDs in license entries are to be enumerated.
- A license file containing only SERVER entries is valid.

Each server entry starts with keyword SERVER and <u>must</u> contain following parameters:

Parameter	Value	Format
HOSTID	The HOSTID string consists of one or more LICMAN server ids, separated by commas	String
HOSTNAME	The HOSTNAME string contains the host name or the IP address of the license server Each host must not be defined more than once. In case of multiple definitions, the last one gains.	String
PORT	TCP port number for communication between LICMAN LLD and LICMAN GLD daemons Please note following port number restrictions: Range 0 to 1023 is reserved for operating system Range 1024 to 49151 is reserved for applications registered by IANA (Internet Assigned Numbers Authority) Range 49152 to 65535 available for unregistered applications, e.g. LICMAN	Number

6.2.2.2. License definition entries

A license definition entry assigns a license to a product.

Notes:

▪ LICENSE entries are supplied by delivery center and must not be modified by customer.

▪ LICENSE entries contained in a network license source are inherited by LLD from GLD.

▪ As LICENSE entries are marked as "node locked" or "floating" and are explicitly assigned to one ore more hosts by LICMAN server id(s), all types of licenses may be contained in one single license file. Each LLD manages its "bode locked" licenses, each GLD manages its "floating" licenses.

Each license entry starts with keyword LICENSE and contains following parameters:

Parameter	Value	Format
KEY	License key of product	Number
NODELOCK	Positional parameter without parameter value The type of license is "node locked". Product is licensed to be used on specified host(s) only.	-
FLOATING	Positional parameter without parameter value The type of license is "floating". License server(s) for this product may run on specified host(s) only.	-
COUNT	Number of concurrent licenses of product An asterisk * indicates an unlimited number of licenses.	Number
STARTS	Starting date of license The date string uses format DD-MMM-YYYY Date is UTC (universal time, coordinated).	Date string

	License becomes valid at beginning of starting date.	
EXPIRES	Expiration date of license The date string uses format DD-MMM-YYYY Date is UTC (universal time, coordinated). License becomes invalid at beginning of expiration date.	Date string
SERIAL	Serial data of license License key and serial data identify a unique license. If multiple licenses with same key and serial data are found, the last one gains. Licenses with same key and different serial data are added.	String
DATA	License data License data usually contain version information. Regarding the license data an application decides whether the license is suitable or not.	String
HOSTID	The HOSTID string consists of one or more LICMAN server ids, separated by commas If license is of type "node locked", it is valid on all hosts matching one of LICMAN server ids. If license is of type "floating" and only 1 LICMAN server id is given, the host matching the LICMAN server id is license server. If license is of type "floating" and more than 1 LICMAN server id is given, all hosts matching the LICMAN server ids are parts of a redundant license server system.	String
COMMENT	License comment An optional comment string may be part of license.	String
SUM	Check sum of license data A check sum is generated from all parameters of a license entry. Only licenses with valid check sum are valid.	String

6.2.2.3. Option definition entries

An option definition entry defines execution parameters. OPTION entries may be supplied by customer.

Notes:

▪ OPTION entries contained in a network license source are <u>not</u> inherited by LLD from GLD. In order to set execution options for LLDs, the license path must specify a license file containing at least one or more OPTION entries.

▪ A license file containing only OPTION entries is valid.

Each option entry starts with keyword OPTION and contains following parameters:

Parameter	Value	Format
DEBUG	Debug output level Default value: 0 0 … No debug output 1 … Debug output without reporting communication details between LICMAN components 2 … Extended debug output including communication details between LICMAN components Even if a LICMAN daemon is not started in debug mode, debug output may be enabled by debug option in license file, debug output is written to message log file.	Number
HEARTBEATFREQ	Heart beat frequency in seconds, a LLD signals to GLD that it is still alive Default value: 120 Minimum value: 60 Maximum value: 1800 A small value increases communication traffic between LLDs and GLD. A large value delays synchronization between LLDs and GLD in case of network problems, unexpected termination of applications, etc.	Number
MSGMAXSIZE	Maximum size of message log file in bytes Default value: 0 If maximum size is 0, no limitation exists. All messages are appended to message log file. If maximum size is greater than 0, the message log file will be cleared if size limit gets reached. Further messages are appended to the empty message log file.	Number
REQBUFSIZE	Buffer size for request log file in bytes Default value: 0 If buffer size is 0, no buffering is used. Each request is immediately written to request log file. If buffer size is greater than 0, request are buffered. If request buffer is full, buffered requests are written to request log file and buffer is cleared.	Number
TCPMAXCONN	Maximum pending GLD socket connections in listening for incoming connections Default value: 1024	Number

	The maximum value depends on operating system. See description of system function "listen".	
TCPCONNTIMEOUT	Maximum wait time in seconds, a LLD is waiting for a connection to additional GLD license servers. Default value: 5 A LLD is waiting a maximum time limited by operating system's TCP timeout value for connection to the first server of a redundant server system and a maximum time of TCPCONNTIMEOUT for additional servers. If TCPCONNTIMEOUT is greater than operating system's TCP timeout value, the internal value is used.	Number
TCPCOMMTIMEOUT	Maximum wait time in seconds, a GLD/LLD is waiting for incoming data Default value: 60 If expected data is not received within maximum wait time, network communication problems are supposed and connection is closed.	Number
THREADCOUNT	Number of GLD threads processing incoming requests concurrently Default value: 10 If value is set to 0, GLD requests are processed sequentially.	Number

6.2.2.4.   License file examples

A license file consists of several lines of variable length.

Example 1:

License file contains one floating license (keyword FLOATING) and one nodelock license (keyword NODELOCK). Floating license with key 4711 allows to run licensed application 10 times concurrently (keyword COUNT). Nodelock license with key 67890 allows to run licensed application unlimited times concurrently (keyword COUNT). One SERVER line is required and defines one LICMAN GLD server.

```
# License file for customer XYZ

OPTION TCPCONNTIMEOUT=30

SERVER HOSTNAME=myhost.mydomain HOSTID=12345ABCDEF PORT=12345

LICENSE KEY=4711 COUNT=10 \
STARTS=1-jul-2006 EXPIRES=01-feb-2007 \
SERIAL="Serial Data 4711" FLOATING HOSTID=12345ABCDEF \
DATA="8.n.n" COMMENT="Product with Key 4711" SUM=\
26653760C13CDE949E8720A7652482F159279C2BC192CA206857248D37\
```

```
141B2D44DE86F042C32124A20322DCCA338AF0DC95CB6D24B3CFE3A164\
F96C82636CE7AA2434C2EDB43A404706FBDC6F8124286A9234A328C253\
8503B87924493B7F9C33F02F5AE408169D24647570E1E591E995CBAFD5\
F42352B588A01723B9B4C20C14A2417661A35AAF0CD069EDACE5324B33\
57120E13D92EEE0505CAD24B92D557F9F8AB4285056A79524AB9EC6A87\
77510A54CBB2448249E1416FEF3443F76A1A15F7C2459BE69585418E08\
04B83474E241A1FBD4F5D0EFF2A8370CD9024BC50055C7149902E58A93\
A7A242CD5EB4553999EAC3825B50A244C6D8EEB84F00D0ED6F1C40023F\
2BC2CE3F85CCF38BE579622431BB2BCE376E0C6E8A5ED6BF231C14D8D7\
C33CC8B9A319148
```

```
LICENSE KEY=67890 COUNT=* \
STARTS=1-jul-2006 EXPIRES=01-feb-2007 \
SERIAL="Serial Data 67890" NODELOCK HOSTID=00AA11BB22CC \
DATA="8.n.n" COMMENT=" Product with Key 67890" SUM=\
24643760C13CDE949E8720A76523CB2E3898B69945C4A1F47CC2436F13\
EEB7982B19D65C6D9D724A20322DCCA338AF0DC95CB6D24B3CFE3A164F\
96C82636CE7AA2434C2EDB43A404706FBDC6F8124286A9234A328C2538\
503B87924493B7F9C33F02F5AE408169D24647570E1E591E995CBAFD5F\
42352B588A01723B9B4C20C14A2417661A35AAF0CD069EDACE5324B335\
7120E13D92EEE0505CAD24B92D557F9F8AB4285056A79524AB9EC6A877\
7510A54CBB2448249E1416FEF3443F76A1A15F7C2459BE69585418E080\
4B83474E2489A056C1D22F3C7817362EED24A0DC9FFD22C196EB2B5744\
6224211E1FED88BACEE369837F0624729FE4FFF8477E56F86EC366241D\
C698F336BB44DC548EF80824A1525690C966F8261DE6DA5024750D728C\
0D9810C9B7068EC42432A45E9E30306966B84C211A242BE9176C07375A\
9E72ACBFC524A5C15DE2D92ADE7911FED954
```

Example 2:

License file contains one floating license (keyword FLOATING) for three redundant servers
(keyword HOSTID contains an enumeration of three host ids). Floating license with key 12345
allows to run licensed application 5 times concurrently (keyword COUNT).Three SERVER lines
are required and define three LICMAN GLD servers.

```
SERVER HOSTNAME=myserver1.mydomain HOSTID=002355FD6D7E PORT=52818
SERVER HOSTNAME=myserver2.mydomain HOSTID=00282ECB8D8 PORT=52818
SERVER HOSTNAME=myserver3.mydomain HOSTID=0028C23D0625 PORT=52818
```

```
LICENSE KEY=12345 COUNT=5 \
STARTS=1-jul-2006 EXPIRES=01-feb-2007 \
SERIAL="Serial Data 12345" FLOATING \
HOSTID=002355FD6D7E,00282ECB8D8,0028C23D0625 \
DATA="8.n.n" COMMENT="Product with Key 12345" SUM=\
26653760C13CDE949E8720A7652482F159279C2BC192CA206857248D37\
141B2D44DE86F042C32124A20322DCCA338AF0DC95CB6D24B3CFE3A164\
F96C82636CE7AA2434C2EDB43A404706FBDC6F8124286A9234A328C253\
8503B87924493B7F9C33F02F5AE408169D24647570E1E591E995CBAFD5\
F42352B588A01723B9B4C20C14A2417661A35AAF0CD069EDACE5324B33\
57120E13D92EEE0505CAD24B92D557F9F8AB4285056A79524AB9EC6A87\
77510A54CBB2448249E1416FEF3443F76A1A15F7C2459BE69585418E08\
04B83474E241A1FBD4F5D0EFF2A8370CD9024BC50055C7149902E58A93\
A7A242CD5EB4553999EAC3825B50A244C6D8EEB84F00D0ED6F1C40023F\
2BC2CE3F85CCF38BE579622431BB2BCE376E0C6E8A5ED6BF231C14D8D7\
C33CC8B9A319148
```

Example 3:

License file contains one nodelock license (keyword NODELOCK) for two clients (keyword HOSTID contains an enumeration of two host ids). Nodelock license with key 55555 allows to run licensed application 3 times concurrently (keyword COUNT) on each of licensed clients. No SERVER line is required.

```
LICENSE KEY=55555 COUNT=3 \
STARTS=1-jul-2006 EXPIRES=01-feb-2007 \
SERIAL="Serial Data 55555" NODELOCK \
HOSTID=002355FD6D7E,00282ECB8D8 \
DATA="8.n.n" COMMENT="Product with Key 55555" SUM=\
26653760C13CDE949E8720A7652482F159279C2BC192CA206857248D37\
141B2D44DE86F042C32124A20322DCCA338AF0DC95CB6D24B3CFE3A164\
F96C82636CE7AA2434C2EDB43A404706FBDC6F8124286A9234A328C253\
8503B87924493B7F9C33F02F5AE408169D24647570E1E591E995CBAFD5\
F42352B588A01723B9B4C20C14A2417661A35AAF0CD069EDACE5324B33\
57120E13D92EEE0505CAD24B92D557F9F8AB4285056A79524AB9EC6A87\
77510A54CBB2448249E1416FEF3443F76A1A15F7C2459BE69585418E08\
04B83474E241A1FBD4F5D0EFF2A8370CD9024BC50055C7149902E58A93\
A7A242CD5EB4553999EAC3825B50A244C6D8EEB84F00D0ED6F1C40023F\
2BC2CE3F85CCF38BE579622431BB2BCE376E0C6E8A5ED6BF231C14D8D7\
C33CC8B9A319148
```

6.2.3.   Working Directory

LICMAN needs a working directory to store following files:

- Internally used files and UNIX sockets
- LICMAN daemons/services need write-access for this working directory and for all files within that directory.

LICMAN applications need read-access for this working directory and for all files within that directory.

If working directory does not exist, LICMAN tries to create the directory and to assign the required permissions.

UNIX

The UNIX working directory is named **/tmp/.licman**

If for some reasons a different working directory should be used, create a symbolic link named **/tmp/.licman** to that existing directory which must have read and write permissions for everyone. Working directory and symbolic link must exist *before* starting LICMAN and LICMAN applications.

It might be necessary to prevent some automatism from deleting the working directory and its contents.

Windows

The Windows temporary directory is determined by following rules:
Temporary directory is set to the value of Windows registry entry
HKLM\Software\T-Systems\Licman\TmpPath or to
%WINDIR%\TEMP, if registry entry not found, or to
%SYSTEMDRIVE%\TEMP, if environment variable **WINDIR** not set, or to
C:\TEMP, if environment variable **SYSTEMDRIVE** not set.
The Windows working directory is built by appending subdirectory **Licman** to the temporary directory.

The Windows environment variable `WINDIR` defines the Windows installation directory; the Windows environment variable `SYSTEMDRIVE` defines the Windows installation drive. `Names of Windows` environment variables are not case sensitive and may be uppercase, lowercase or mixed case.

If for some reasons a different working directory should be used, create a symbolic link also known as *junction* using Windows utility LINKD from working directory name built as described above to that existing directory which must have read and write permissions for everyone. Working directory and symbolic link must exist *before* starting LICMAN and LICMAN applications. Utility *LINKD.EXE* is part of *Windows 2003 Resource Kit Tools*.

6.2.4.   Log File Directory

LICMAN needs a log file directory to store following files:

▪   Message log files

▪   Request log files (optional)

LICMAN daemons/services need write-access for this working directory and for log files within that directory.

If log file directory is not specified, working directory is used instead.

If log file directory is specified but does not exist, LICMAN tries to create the directory and to assign the required permissions.

Setting UNIX Log File Directory

The UNIX log file directory is specified by environment variable `LICMAN_LOG_PATH`.

Setting Windows Log File Directory

The Windows log file directory is specified by registry entry
`HKLM\Software\T-Systems\Licman\LogPath`

Alternatively license path may be specified UNIX-like by environment variable (see above). If both environment variable and registry key are set, environment variable takes priority over registry key.

6.2.5.   Message Log File

A message log file is located in LICMAN log file directory.
GLD's message log file is named `gld.<hostname>.message.log,`
LLD's message log file is named `lld.<hostname>.message.log`,
where `<hostname>` is the current host's name.

It contains a protocol of all errors and messages written by GLD or LLD. Occasionally the log file should be reviewed and its contents deleted.

6.2.6.   Request Log File

A request log file is located in LICMAN log file directory.
GLD's request log file is named `gld.<hostname>.request.log`,
LLD's request log file is named `lld.<hostname>.request.log`,
where `<hostname>` is the current host's name.

Each GLD and LLD keeps track of its own allocation and free requests. A GLD request log file reports allocation and free requests of floating licenses managed by GLD, a LLD request log file reports allocation and free requests of node locked licenses managed by LLD.

To enable logging of requests, set bit 0 of LICMAN execution flags to 1.

Each allocation and release request creates one record, which will be appended to the end of the requests file. All records of the requests file are in chronological order.

Each record has a length of 94 characters and holds the following information:

Column	Keyword	Description
01-08	Date	Current date, format YY/MM/DD
10-17	Time	Current time, format hh:mm:ss
19-26	Sec	Time of license allocation in seconds, hex format
27-31	Msec	Time of license allocation in microseconds, hex format
		Above time values are UTC (universal time, coordinated) since January 1, 1970
		IP address of the requesting host
33-47	Host	Request type (0: license freed / 1: license allocated / 2: allocation denied)
49-49	Type	
		User name of the requesting user
51-58	User	Process number of the requesting process
60-67	Pid	License key number of the requested license according to the license file
69-76	Key	
		Total number of licenses for this module
78-85	Total	(-1: unlimited number)
		Number of licenses used after performing allocation/free request
87-94	Used	

6.3.   Execution Flags

Flags are used to control execution of LICMAN daemons/services.

The flag values are single bits of a bit mask. Bits are numbered from 0 to 31, where bit 0 is the least significant bit and 31 is the most significant bit. Bit 0 has a decimal value of $1=2^0$, bit 1 has a value of $2=2^1$, …, bit n has a value of 2^n.

The value of the bit mask is the addition of the single bit values. If for example bits 0 and 4 are set, set value of the bit mask is 2^0+2^4 = 1+16 = 17.

Currently only 1 execution flag bit is used.

Flag bit	Description
0	Enable logging of license allocation and free requests in request log file
Default: bit not set

Setting UNIX Execution Flags

On UNIX operating systems, a starting GLD and/or LLD daemon reads execution flags from environment variable `LICMAN_FLAGS`. Execution flags remain active while GLD and/or LLD daemon is running.

Setting Windows Execution Flags

On Windows operating systems, a starting GLD and/or LLD daemon reads execution flags from registry key
`HKEY_LOCAL_MACHINE\SOFTWARE\T-Systems\Licman\Flags`
Alternatively execution flags may be specified UNIX-like by environment variable (see above). If both environment variable and registry key are set, environment variable takes priority over registry key.

Execution flags remain active while GLD and/or LLD daemon is running.

7.   LICMAN Utilities

7.1.   LICMAN Test Utility

Before using LICMAN test utility, make sure

- that LICMAN LLD process is running on current host
- that LICMAN GLD process is running on each licensed server. In case of a redundant server system with 3 servers, LICMAN GLD must be running on at least 2 servers.

The LICMAN test utility can be used to verify a LICMAN installation and to list all currently used licenses.

UNIX: `<installation_directory>/licman20_appl`

Windows: Program Files -> T-Systems -> Licman 2.0 -> Licman Test

Following dialog appears:

```
LICMAN test utility


 0 Free     license key
 1 Alloc    license key
 2 Query    allocated licenses
 3 Enable   license verification
 4 Query    available licenses
 X Quit

Menu option ?
```


Option 0:

Use option 0 to free an allocated license. Enter a product key number you have already allocated with option 1 of this test utility.

Option 1:

Use option 1 to allocate a license. Enter a product key number as contained in your license file(s).

Return code will be 0 if the LICMAN test utility successfully allocated the license.

Otherwise, check the error message returned by the LICMAN test utility, check message log file(s) for additional error messages and solve the problem causing the error message (see chapter 9 "Troubleshooting").

Option 2:

Option 2 lists all licenses currently allocated by LLD and GLDs known to the LLD.
For each allocated license you get one line of information:

Date	Licenses allocation date
Time	Licenses allocation time
User	User name

Key	License key number
Userid	User id
Procid	Process id of the application using this allocated license
Seq	GLD sequence number if redundant GLDs are used. Will be 0 if no redundant GLDs are used.
Client/Server	Client host name / server host name

Example 1 – no redundant servers:

```
Date     Time     User     Key  Userid  Procid Seq Serial      Client/Server
09/11/23 09:33:38 demo     101    2060   21129   0 SER001      pc123/srv01
09/11/26 15:02:20 test     101     508    6642   0 SER001      ws007/srv01
09/11/23 09:33:38 demo     111    2060   21129   0 SER001      pc123/srv01
09/11/26 15:02:20 test     111     508    6642   0 SER001      ws007/srv01
```

Example 2 – three redundant servers:

```
Date     Time     User     Key  Userid  Procid Seq Serial      Client/Server
09/11/23 09:33:38 demo    1111    2060   21129   0 SER001      pc123/srv01
09/11/23 09:33:38 demo    1111    2060   21129   1 SER001      pc123/srv02
09/11/23 09:33:38 demo    1111    2060   21129   2 SER001      pc123/srv03
```

Option 3:

Use option 3 to enable/disable license verification. If license verification is enabled, license serial data string and license data string will be displayed, when trying to allocate a license. Depending on the data shown, allocation request may get confirmed or discarded.

Option 4:

Option 4 lists all licenses available by LLD and GLDs known to the LLD.
For each available license you get information as shown:

```
License   5:
------------
Product key        : 12345
Number of Licenses : unlimited
In use             : 1
Type               : floating
Start date         : 01-Jul-2006
End date           : 01-Feb-2007
Serial information : Serial data
Version information : 10.n.n
Comment            : Comment data
Server name(s)     : myhost.mydomain
Server address(es) : 192.168.1.1
```

Option X:

Option X quits the LICMAN test utility.

UNIX

LICMAN test utility `licman20_appl` may be used without interactive dialog. Use a UNIX shell script like the following, supplying the test utility with input:

```
#!/bin/sh
<installation_directory>/licman20_appl <<@@ 1>/dev/null
2
x
@@
```

or create a text file named `input.txt` containing 1 input to the test utility per line and run

```
<installation_directory>/licman20_appl <input.txt 1>/dev/null
```

Windows

LICMAN test utility `licman20_appl.exe` may be used without interactive dialog. Create a text file named `input.txt` containing 1 input to the test utility per line, open a console window and run

```
<installation_directory>/licman20_appl.exe <input.txt 1>NUL
```

7.2.   LICMAN Id Utility

The LICMAN id utility supplies LICMAN system id(s) required

- to license a product with a node locked license managed by a LICMAN LLD on a specific host

- to license a product with a floating license managed by a LICMAN GLD on a specific host

You get the LICMAN system id(s) by running the LICMAN id utility

UNIX: `<installation_directory>/licman20_id`

Windows: Program Files -> T-Systems -> Licman 2.0 -> Licman Id

The LICMAN system id is required by the T-Systems delivery department for creating a valid license file for your desired Workstation.

Output example of LICMAN id utility

- **for operating system AIX**
  ```
  Valid LICMAN system id(s) found:
  00513ADF5D00
  ```

- **for operating system HP-UX**
  ```
  Valid LICMAN system id(s) found:
  2005487395
  ```

- **for operating system IRIX**
  ```
  Valid LICMAN system id(s) found:
  1762679270
  ```

- **for operating system SunOS**
  ```
  Valid LICMAN system id(s) found:
  83cdcf82
  ```

- **for operating system Linux**
  ```
  Looking for network adapter cards ...

  Adapter 1:
  Interface        = eth0
  Hardware address = 00:E0:81:61:86:B0

  Adapter 2:
  Interface        = eth1
  Hardware address = 00:0C:6E:51:57:E0
  ```

```
Valid LICMAN system id(s) found:
00E0816186B0 000C6E5157E0
```

- **for operating system Windows**
  ```
  Looking for network adapter cards ...

  Adapter 1:
  Name              = {906CB3F6-7DED-47CF-8C95-5018F17D1840}
  Description       = Broadcom NetXtreme Gigabit Ethernet
  Physical address  = 00-E0-81-61-86-B0

  Adapter 2:
  Name              = {763B914B-0D91-43D8-89F6-52AF7B21D01F}
  Description       = 3Com Gigabit LOM (3C940)
  Physical address  = 00-0C-6E-51-57-E0

  Valid LICMAN system id(s) found:
  00E0816186B0 000C6E5157E0
  ```

8. Compatibility with previous LICMAN versions

LICMAN 2.0 is not compatible to previous LICMAN versions. However this incompatibility concerns only communication between GLD and LLD components of LICMAN, i.e. a LICMAN 2.0 LLD cannot communicate with a LICMAN 1.2 GLD, a LICMAN 1.2 cannot communicate with a LICMAN 2.0 GLD.

As only 1 LICMAN GLD and/or 1 LICMAN LLD is allowed to run on one workstation, it is not possible

- a LICMAN 2.0 GLD and a LICMAN 1.2 GLD running the same time
- a LICMAN 2.0 LLD and a LICMAN 1.2 LLD running the same time

Trying to start a second daemon will fail. The second daemon will find a first daemon running and terminate itself.

In order to successfully allocate a license, a **LICMAN 2.0 application** requires a **LICMAN 2.0 LLD** to be running.

In order to successfully allocate a license, a **LICMAN 1.2 application** requires

- either a LICMAN 1.2 LLD to be running
- or a LICMAN 2.0 LLD to be running in LICMAN 1.2 compatibility mode

8.1. Setting up LICMAN 1.2 Compatibility Mode

In order to start up in LICMAN 1.2 compatibility mode, a LICMAN 2.0 LLD must be supplied with a LICMAN 1.2 license file.

As a LICMAN 1.2 application communicates to its LLD via socket, a LLD port number is defined in each LICMAN 1.2 license file. The only data read by a LICMAN 1.2 application is the LLD port number. This port number is required to connect to LLD.

The way a LICMAN 2.0 communicates to its LLD is different from LICMAN 1.2; no LLD port number is required. Therefore LICMAN 2.0 applications do not access a license file.

If a LICMAN 2.0 LLD is supplied with a LICMAN 1.2 license file, it looks for the LICMAN 1.2 LLD port number and, in addition to the channel for incoming LICMAN 2.0 application requests, it opens a second channel using that port number for incoming LICMAN 1.2 application requests. Communication over first channel uses the newer LICMAN 2.0 protocol; communication over second channel uses the older LICMAN 1.2 protocol.

As both, LICMAN 1.2 application and LICMAN 2.0 LLD only look for LLD port number, a valid compatibility license file named `licenses` defining port number **52817** needs 6 lines and might look like follows:

```
Line    Contents of line
 1    Licman 1.2/2.0 compatibility license file
 2    Check sum not used
 3    01.01.01 # Start of license not used
 4    02.02.02 # Last update not used
 5    03.03.03 # End of license not used
 6    52817 0
```

Notes:

- First and only word read is word 1 of line 6, the LLD port number, other lines are skipped.
- Make sure that LLD port number is different to PORT value in any LICMAN 2.0 license file defining commiunication port to GLD.

- Make sure that LLD port number is not used by any other application on your client.

Location of compatibility license file `licenses` is announced to LICMAN 1.2 applications and LICMAN 2.0 LLD according to the rules of LICMAN 1.2.

UNIX

LICMAN 1.2 license path is set by environment variable `debisLICDIR`. Set value of environment variable to the license file's directory path name.

Some T-Systems products are using a default directory, if environment variable `debisLICDIR` is not set. For example, the T-Systems product MEDINA uses `<installation_directory>/cae/prod/data/licman` as default license directory, where `<installation_directory>` is product's installation directory.

Note:

Compatibility mode may be easily enabled during LICMAN daemon startup by startup script `licman20.` Startup script creates a compatibility license file `licenses` in working directory `/tmp/.licman`. To enable compatibility mode, set environment variable `debisLICDIR` to `/tmp/.licman` *before* calling startup script `licman20` and *before* running applications developed using LICMAN 1.2. Application versions older than January 1st, 2008 usually are concerned.

Windows

LICMAN 1.2 license path is defined by registry key
`HKEY_LOCAL_MACHINE\SOFTWARE\T-Systems\Licman\LICPATH`
Set value of registry key to the license file's directory path name.

If registry key does not exist, license file is searched in directory
`%debisCAE%\cae\prod\data\licman` instead and environment variable `debisCAE` must
exist. This behavior is for compatibility with LICMAN 1.2 applications running on Windows,
which use exactly same method for detecting a valid license file. Please note, environment
variable `debisLICDIR` is not used and has never been used by LICMAN running on Windows.

Note:

Compatibility mode may be easily enabled during LICMAN for Windows installation. This
includes creation of

- registry key
  `HKEY_LOCAL_MACHINE\SOFTWARE\T-Systems\Licman\LICPATH`

- compatibility license file
  `%CommonProgramFiles%\T-Systems\Licman\licenses`

8.2.   Migrating from LICMAN 1.2 to LICMAN 2.0

If you need to run older LICMAN 1.2 licensed products and newer LICMAN 2.0 licensed
products concurrently on same workstations, migrating from LICMAN 1.2 to LICMAN 2.0 is
required.

Following steps have to be carried out:

- Get and provide a new LICMAN 2.0 license file containing licenses for all product previously
  licensed by old LICMAN 1.2 license file

- Provide a compatibility license file as specified or use your existing LICMAN 1.2 license file
  as compatibility license file (see chapter 8.1 "General Remarks")

- Stop LICMAN 1.2 (see chapter 3.3 "Stop License Manager")

- Disable LICMAN 1.2 (see chapter 3.4 "Disable/uninstall previous LICMAN 1.2")

- Install LICMAN 2.0 if not already installed

- UNIX only: Find and replace following LICMAN 1.2 executables and scripts by LICMAN 2.0
  executables and scripts in your existing installations:
  Copy LICMAN 2.0 GLD executable licman20_gld to LICMAN 1.2 GLD executable
  licman12_gld
  Copy LICMAN 2.0 LLD executable licman20_lld to LICMAN 1.2 LLD executable
  licman12_lld
  Copy LICMAN 2.0 LLD startup script licman20 to LICMAN 1.2 LLD startup script licman12,
  if existent

- UNIX only: Modify your own existing LICMAN startup mechanisms to additionally provide a
  LICMAN 2.0 license path (see chapter 6.2.1 "License Path") and to start LICMAN 2.0
  executables instead of LICMAN 1.2 ones

- Start LICMAN 2.0 (see chapter 5.1 "Starting LICMAN ")

It may take a considerable amount of time to migrate a large number of client installations from
LICMAN 1.2 to LICMAN 2.0. If LICMAN GLD license server has been migrated to LICMAN 2.0,
clients will get no longer floating licenses until they are migrated to LICMAN 2.0 too. If it is not
acceptable that clients not yet migrated to LICMAN 2.0 will get no licenses during migration
period following procedure is recommended:

- Preserve existing LICMAN 1.2 installation running on license server(s) while client migration
  to LICMAN 2.0 is in progress

- Setup a temporary alternate license server or a set of alternate redundant license servers with LICMAN 2.0. Note: licenses for alternate server(s) valid for a limited migration period are required. Ask delivery center for temporary licenses.

- Migrate client by client from LICMAN 1.2 to LICMAN 2.0. While clients not migrated still receive licenses from existing LICMAN 1.2 license server(s), clients already migrated receive licenses from alternate LICMAN 2.0 license server(s)

- Once all clients are migrated to LICMAN 2.0, replace LICMAN 1.2 server(s) installation by LICMAN 2.0

- Set license path on clients to license file containing licenses for proper license server(s)

9.   Troubleshooting

This chapter tries to explain some of the most frequent error messages and to help solving the problem. The LICMAN test program may report error messages. Additional error messages, e.g. operating system dependant messages, may be found in the message log file in the working directory.

```
Can't create directory <directory>
Can't access directory <directory>
```

Meaning:

- Working directory does not exist and LICMAN is not able to create working directory
- A path component of working directory does not exist
- Object with same name exists but is no directory

Solution:

- Correct permissions of path components
- Create missing path component(s)
- Remove object with same

```
Can't open license file <path>
```

Meaning:

- License file does not exist
- No read permission for license file

Solution:

- Create license file
- Correct permissions of license file

```
Can't open message log file <path>
Can't open request log file <path>
Can't open path file <path>
```

Meaning:

- Working directory does not exist
- No write permission for working directory
- No write permission for file

Solution:

- Prevent working directory to get deleted
- Prevent file permissions to get changed
- Correct permissions

`Can't get license path`

Meaning:

- License path not defined

Solution:

- Set environment variables defining license path before starting LICMAN (UNIX/Windows)
- Set registry keys defining license path before starting LICMAN (Windows)

`License path contains invalid file entry <path>`

Meaning:

- License file defined in license path not found
- No read permission for license file

Solution:

- Create license file
- Correct permissions of license file

`License path contains invalid server entry <entry>`

Meaning:

- Syntax of network license source invalid
- Unable to evaluate IP address of host name

Solution:

- Correct license path definition before starting LICMAN
- Assign IP address to host name

`License data parse error`

Meaning:

- License file contains invalid entry

Solution:

- Correct license file SERVER and/or OPTION entries
- Get license file with valid LICENSE entries

```
License contains invalid sum value
License contains no sum value
```

Meaning:

- License file contains invalid LICENSE entry

Solution:

- Get license file with valid LICENSE entries
- Avoid modifying LICENSE entries

```
Server data incomplete
```

Meaning:

- License file contains incomplete SERVER entry

Solution:

- Supply SERVER entry containing all required keywords HOSTNAME, HOSTID and PORT

```
Can't find valid system id
```

Meaning:

- Unable to evaluate LICMAN system id(s) of this host

Solution:

- Activate network adapter card(s) on Linux or Windows PC

```
Unable to background Licman GLD process
Unable to background Licman LLD process
```

Meaning:

- Unable to switch started process do background daemon (UNIX only)

Solution:

- See error message log for additional information

```
Can't initialize service
```

Meaning:

- Unable to change service state (Windows only)

Solution:

- See error message log for additional information

```
Licman GLD killed by signal <signal>
Licman LLD killed by signal <signal>
```

Meaning:

- LICMAN process was forced to terminate due to an external signal
- LICMAN process terminated due to an internal error condition

Solution:

- If LICMAN terminated due to an internal error condition, report problem do T-Systems hotline.

```
License key <key> still allocated by application
License key <key> still allocated by process with id <pid>
Stop application before restarting Licman LLD
```

Meaning:

- An unknown application process has still allocated a license with product key as shown
- An application process with known process id has still allocated a license with product key as shown

Solution:

- LICMAN LLD cannot restart before all applications with an allocated license are terminated. Wait for application termination before restarting LLD.

```
Output device invalid, can't start in debug mode
```

Meaning:

- Try to start LICMAN service, while debug mode is enabled (Windows only)

Solution:

- Disable debug mode and restart LICMAN service

```
Port <port> already opened for applications
```

Meaning:

- LICMAN LLD is running LICMAN 1.2 compatibility mode. Port number is doubly defined:
  - as port number for incoming LICMAN 1.2 application requests in LICMAN 1.2 compatibility license file
  - as port number for outgoing GLD server requests in SERVER entry of license file

Solution:

- Use different port numbers for incoming and outgoing requests

`No network address for server <hostname> found`

Meaning:

- LICMAN LLD is not able to evaluate IP address for host

Solution:

- Check host names in license path
- Check host names of SERVER entries in license file(s)
- Assign IP address to host name

`Opening channel for applications failed`

Meaning:

- LICMAN LLD is not able to establish channel for incoming application requests

Solution:

- Check if port number is already in use by other process
- See error message log for additional information

`Opening channel to Licman GLD on server <hostname> failed`

Meaning:

- LICMAN LLD is not able to establish channel for outgoing GLD server requests

Solution:

- Check if port number is already in use by other process
- See error message log for additional information

`Opening channel for Licman LLDs failed`

Meaning:

- LICMAN GLD is not able to establish channel for incoming LLD clients requests

Solution:

- Check if port number is already in use by other process
- See error message log for additional information

No valid port number to Licman GLD on server <hostname>

Meaning:

- No port number or no valid port number assigned to LLD for outgoing GLD server requests

Solution:

- Check if port number is assigned by SERVER entry of license file for this server
- Check if port number is within valid range

No valid port for Licman LLDs

Meaning:

- No port number or no valid port number assigned to GLD for incoming LLD clients requests

Solution:

- Check if port number is assigned by SERVER entry of license file for this server
- Check if port number is within valid range

System time on client <hostname> differs more than 24 hours

Meaning:

- A license allocation request from LLD client shown was received by GLD. The LLD's system time differs more than 24 hours from GLD's system time. Times in UTC (universal time, coordinated) are compared. The allocation request is refused.

Solution:

- Correct GLD server's and LLD client's internal clock
- Correct GLD server's and LLD client's time zones

Application communication error

Meaning:

- A communication error between application and LLD occurred

Solution:

- See error message log for additional information

`Invalid data from application`

Meaning:

▪ Data received from application differs from data expected

Solution:

▪ See error message log for additional information

`Licman GLD on server <hostname> communication error`
`Licman LLD on client <hostname> communication error`

Meaning:

▪ A communication error between LLD client and GLD server occurred

Solution:

▪ See error message log for additional information

`Invalid data from Licman LLD on client <hostname>`
`Invalid data from Licman GLD on server <hostname>`

Meaning:

▪ Data received from LLD/GLD differs from data expected

Solution:

▪ See error message log for additional information

`Incorrect data length`

Meaning:

▪ Size of a data packet received or sent by a LICMAN component differs from data packet size expected

Solution:

▪ See error message log for additional information

10. Change Log

10.1. Version 2.0.1

26-Sep-2006

- Initial release as β version

10.2. Version 2.0.2

12-Dec-2006

- GLD/LLD no longer require license file(s) to be writeable

10.3. Version 2.0.3

25-Jan-2007

- GLD/LLD/ID allow new AIX 64-bit system id as LICMAN system id, if available

10.4. Version 2.0.4

05-Feb-2007

- New options THREADCOUNT and HEARTBEATFREQ
- Existing environment variables take priority over registry keys on Windows
- Identify client at server by system id if IP address changed
- Thread-safe request logging

10.5. Version 2.0.5

07-Feb-2007

- Remove trailing / in path delivered by Windows system function GetTempPath
- Identify client at server first by system id then by IP address

10.6. Version 2.0.6

14-Feb-2007

- Work around startup problem because of SuSE 9.2 64-bit Linux fcntl bug

10.7. Version 2.0.7

19-Mar-2007

- Modified determination of LICMAN working directory according to customer needs on Windows

10.8. Version 2.0.8

05-Apr-2007

- Performance improvements in communication between GLD and LLD

17-Oct-2007

- Added notice about possibility to use symbolic link as LICMAN working directory to LICMAN User's Manual

10.9.   Version 2.0.9

29-Oct-2007

- Check for error conditions
  No valid port number to Licman GLD on server <hostname>
  No valid port for Licman LLDs

10.10.   Version 2.0.10

12-Dec-2007

- Avoid GLD crash on SUN Solaris 10 operating system
- Print LICMAN version if debug is enabled

10.11.   Version 2.0.11

08-Jan-2008

- Allow net devices with ROOT enumeration on Windows operating system when searching for a valid LICMAN ID

10.12.   Version 2.0.11a

22-Jan-2008

- Added switch to enable LICMAN 1.2/LICMAN 2.0 compatibility mode during LICMAN for Windows installation
- Preparing LICMAN 1.2/LICMAN 2.0 compatibility mode for UNIX if LICMAN daemons are started by startup script licman20

10.13.   Version 2.0.12

30-Jan-2008

- Ignore net devices of type different from type "Ethernet" on Linux operating system when searching for a valid LICMAN ID

10.14.   Version 2.0.13

15-Feb-2008

- Allow customization of log file directory

10.15.   Version 2.0.15

10-Apr-2008

- Skip version number 2.0.14
- Badly conditioned DNS might occupy all GLD threads leading to malfunction. Avoid DNS name resolution timing problem. Therefore always print IP address instead of host name to request log file.

- Add timestamp to GLD and LLD debug output

10.16. Version 2.0.16

17-Apr-2008

- Repair error introduced with GLD version 2.0.15: Querying licenses "in use" did not show correct information
- Repair possible malfunction in GLD because of concurrent thread processing

10.17. Version 2.0.17 / 2.0.17a

18-Apr-2008

- Repair possible malfunction in GLD running in debug mode because of concurrent thread processing

16-Jul-2008

- Add "Silent Installation" possibility on Windows

10.18. Version 2.0.18

02-Dec-2008

- Avoid potential GLD crash occurring under certain circumstances in case of an error condition during communication with LLD.

10.19. Version 2.0.19

30-Nov-2009

- Avoid default working directory %WINDIR%/Temp access problems because of special permissions assigned to this directory in Windows Vista and newer
- Skip invalid license items with time values < 0 to avoid crash in LICMAN application utility on Windows platform
- Allow LLD running as "Detached Process" on Windows with normal user permissions

10.20. Version 2.0.20

21-Jan-2013

- Correct process numbers of GLD and LLD process written to "pid" file within working directory on Windows
- Correct counting of allocated licenses on redundant license servers i.e. GLDs, if allocation request is granted by some servers while denied by at least one redundant license server

T **·** **·Systems**