

# Leda

## Installation Guide

---

Version 2006.06  
June 2006



Comments?  
E-mail your comments about this manual to  
[leda-support@synopsys.com](mailto:leda-support@synopsys.com).

**SYNOPSYS**<sup>®</sup>

## Copyright Notice and Proprietary Information

Copyright © 2005 Synopsys, Inc. All rights reserved. This software and documentation contain confidential and proprietary information that is the property of Synopsys, Inc. The software and documentation are furnished under a license agreement and may be used or copied only in accordance with the terms of the license agreement. No part of the software and documentation may be reproduced, transmitted, or translated, in any form or by any means, electronic, mechanical, manual, optical, or otherwise, without prior written permission of Synopsys, Inc., or as expressly provided by the license agreement.

### Destination Control Statement

All technical data contained in this publication is subject to the export control laws of the United States of America. Disclosure to nationals of other countries contrary to United States law is prohibited. It is the reader's responsibility to determine the applicable regulations and to comply with them.

### Disclaimer

SYNOPSYS, INC., AND ITS LICENSORS MAKE NO WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, WITH REGARD TO THIS MATERIAL, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

### Registered Trademarks (®)

Synopsys, AMPS, Arcadia, C Level Design, C2HDL, C2V, C2VHDL, Cadabra, Calaveras Algorithm, CATS, CSim, Design Compiler, DesignPower, DesignWare, EPIC, Formality, HSPICE, Hypermodel, iN-Phase, in-Sync, Leda, MAST, Meta, Meta-Software, ModelAccess, ModelTools, NanoSim, OpenVera, PathMill, Photolynx, Physical Compiler, PowerMill, PrimeTime, RailMill, Raphael, RapidScript, Saber, SiVL, SNUG, SolvNet, Stream Driven Simulator, Superlog, System Compiler, Testify, TetraMAX, TimeMill, TMA, VCS, Vera, and Virtual Stepper are registered trademarks of Synopsys, Inc.

### Trademarks (™)

abraCAD, abraMAP, Active Parasitics, AFGen, Apollo, Apollo II, Apollo-DPII, Apollo-GA, ApolloGAI, Astro, Astro-Rail, Astro-Xtalk, Aurora, AvanTestchip, AvanWaves, BCView, Behavioral Compiler, BOA, BRT, Cedar, ChipPlanner, Circuit Analysis, Columbia, Columbia-CE, Comet 3D, Cosmos, CosmosEnterprise, CosmosLE, CosmosScope, CosmosSE, Cyclelink, Davinci, DC Expert, DC Expert Plus, DC Professional, DC Ultra, DC Ultra Plus, Design Advisor, Design Analyzer, Design Vision, DesignerHDL, DesignTime, DFM-Workbench, DFT Compiler, Direct RTL, Direct Silicon Access, Discovery, DW8051, DWPCI, Dynamic-Macromodeling, Dynamic Model Switcher, ECL Compiler, ECO Compiler, EDANavigator, Encore, Encore PQ, Evaccess, ExpressModel, Floorplan Manager, Formal Model Checker, FoundryModel, FPGA Compiler II, FPGA Express, Frame Compiler, Galaxy, Gattran, HDL Advisor, HDL Compiler, Hercules, Hercules-Explorer, Hercules-II, Hierarchical Optimization Technology, High Performance Option, HotPlace, HSPICE-Link, iN-Tandem, Integrator, Interactive Waveform Viewer, i-Virtual Stepper, Jupiter, Jupiter-DP, JupiterXT, JupiterXT-ASIC, JvXtreme, Liberty, Libra-Passport, Library Compiler, Libra-Visa, Magellan, Mars, Mars-Rail, Mars-Xtalk, Medici, Metacapture, Metacircuit, Metamanager, Metamixsim, Milkyway, ModelSource, Module Compiler, MS-3200, MS-3400, Nova Product Family, Nova-ExploreRTL, Nova-Trans, Nova-VeriLint, Nova-VHDLint, Optimum Silicon, Orion\_ec, Parasitic View, Passport, Planet, Planet-PL, Planet-RTL, Polaris, Polaris-CBS, Polaris-MT, Power Compiler, PowerCODE, PowerGate, ProFPGA, ProGen, Prospector, Protocol Compiler, PSMGen, Raphael-NES, RoadRunner, RTL Analyzer, Saturn, ScanBand, Schematic Compiler, Scirocco, Scirocco-i, Shadow Debugger, Silicon Blueprint, Silicon Early Access, SinglePass-SoC, Smart Extraction, SmartLicense, SmartModel Library, Softwire, Source-Level Design, Star, Star-DC, Star-MS, Star-MTB, Star-Power, Star-Rail, Star-RC, Star-RCXT, Star-Sim, Star-SimXT, Star-Time, Star-XP, SWIFT, Taurus, Taurus-Device, Taurus-Layout, Taurus-Lithography, Taurus-Process, Taurus-Topography, Taurus-Visual, Taurus-Workbench, TimeSlice, TimeTracker, Timing Annotator, TopoPlace, TopoRoute, Trace-On-Demand, True-Hspice, TSUPREM-4, TymeWare, VCS Express, VCSi, Venus, Verification Portal, VFormal, VHDL Compiler, VHDL System Simulator, VirSim, and VMC are trademarks of Synopsys, Inc.

### Service Marks (SM)

MAP-in, SVP Café, and TAP-in are service marks of Synopsys, Inc.

SystemC is a trademark of the Open SystemC Initiative and is used under license.

ARM and AMBA are registered trademarks of ARM Limited.

All other product or company names may be trademarks of their respective owners.

# Contents

<b>Preface</b> .....	<b>5</b>
About This Manual .....	5
Related Documents .....	5
Manual Overview .....	5
Typographical and Symbol Conventions .....	6
Getting Leda Help .....	7
The Synopsys Web Site .....	7
<b>Chapter 1</b>	
<b>Installing Leda</b> .....	<b>9</b>
Introduction .....	9
Pre-Installation Requirements and Setup .....	9
Global and Local Installation .....	10
Hardware/OS Requirements .....	10
Loading Leda .....	11
Global Installation (Required) .....	12
Extracting Files .....	12
Installing and Updating the Licenses .....	15
License File Variable Settings .....	16
Retrieving License Use Information .....	16
Local Installation (Optional) .....	17
Configuring the Checker .....	18
LEDA_CONFIG .....	18
LEDA_RESOURCES (VHDL only) .....	19
LEDA_HTML_DOC_PATH and LEDA_HTML_USR_PATH .....	19
HTML_NAVIGATOR .....	20
LEDA_READER .....	20
Running the Post-Installation Test .....	21
<b>Index</b> .....	<b>23</b>



---

# Preface

---

## About This Manual

This guide explains how to install, configure, and test the Leda Specifier and Checker tools.

## Related Documents

This manual is part of the Leda documentation set. To see a complete listing, refer to the [Leda Document Navigator](#).

## Manual Overview

This manual contains the following chapters:

Preface

Describes the manual and lists the typographical conventions and symbols used. Explains how to get technical assistance.

[Chapter 1](#)  
“[Installing Leda](#)”

Detailed instructions for downloading, installing, configuring, and testing the Leda Specifier and Checker tools.

## Typographical and Symbol Conventions

The following conventions are used throughout this document:

**Table 1: Documentation Conventions**

Convention	Description and Example
%	Represents the UNIX prompt.
<b>Bold</b>	User input (text entered by the user). % <b>cd</b> \$LMC_HOME/hd1
Monospace	System-generated text (prompts, messages, files, reports). No Mismatches: 66 Vectors processed: 66 Possible"
<i>Italic or Italic</i>	Variables for which you supply a specific value. As a command line example: % <b>setenv</b> LMC_HOME <i>prod_dir</i> In body text: In the previous example, <i>prod_dir</i> is the directory where your product must be installed.
(Vertical rule)	Choice among alternatives, as in the following syntax example: -effort_level low   medium   high
[ ] (Square brackets)	Enclose optional parameters: <i>pin1</i> [ <i>pin2</i> ... <i>pinN</i> ] In this example, you must enter at least one pin name ( <i>pin1</i> ), but others are optional ([ <i>pin2</i> ... <i>pinN</i> ]).
<b>TopMenu &gt; SubMenu</b>	Pulldown menu paths, such as: <b>File &gt; Save As ...</b>

## Getting Leda Help

For help with Leda, send a detailed explanation of the problem, including contact information, to [leda-support@synopsys.com](mailto:leda-support@synopsys.com).

## The Synopsys Web Site

General information about Synopsys and its products is available at this URL:

<http://www.synopsys.com>





---

# 1

## Installing Leda

---

### Introduction

This chapter explains how to install, configure, and test the Leda Specifier and Checker tools. The information is presented in the following sections:

- [“Pre-Installation Requirements and Setup” on page 9](#)
- [“Global and Local Installation” on page 10](#)
- [“Hardware/OS Requirements” on page 10](#)
- [“Loading Leda” on page 11](#)
- [“Global Installation \(Required\)” on page 12](#)
- [“Local Installation \(Optional\)” on page 17](#)
- [“Configuring the Checker” on page 18](#)
- [“Running the Post-Installation Test” on page 21](#)

### Pre-Installation Requirements and Setup

Before installing the Leda software, you need to obtain a license from Synopsys. Contact your sales rep if you need assistance.

The root directory where Leda is installed is referred to by the environment variable `LEDA_PATH`. Set this variable before installing the software, as shown in the following example:

```
% setenv LEDA_PATH installation_dir
```

## Global and Local Installation

When you install Leda, the Specifier and Checker tools are both installed in the target directory. The licenses you purchased from Synopsys determine which tools you can use. The Specifier tool is licensed separately and includes the Checker. The Checker tool is also licensed separately as a standalone product. The Checker GUI looks just like the Specifier GUI. The only difference is the absence of a Policy Manager in the Leda Rule Wizard. This means that you cannot write and compile your own rules if you only have a Checker license.

There are two ways to install the Leda software. Global installation is required and is the only way to install Leda if you only have a Checker license. If you have a Specifier license, you also install the Leda software globally, and then have the option to install locally if you want to add or modify rules for your own use without affecting other users at your site who are referencing the global installation.

- **Global Installation**—You or a system administrator installs the software in a location that cannot be overwritten by other users. See [“Global Installation \(Required\)” on page 12](#).
- **Local Installation**—Here, the executables are referenced from the global installation, but you can add or modify rules at a local level. Therefore, you must complete the global installation first, before installing locally. See [“Local Installation \(Optional\)” on page 17](#).

## Hardware/OS Requirements

To install and use Leda, you must be running one of the following operating systems:

- Sun Solaris 2.8 or 2.9
- HP-UX 11.0
- Red Hat Linux 3.0 on x86 architecture (32-bit)
- IBM - AIX 4.3.3 or 5.1
- AMD Opteron 64 bit
- Suse32 on Intel Xeon - EM64T processors.
- Suse64 on Intel Xeon - EM64T processors.
- Linux RHEL 3.0 Opteron AMD 64-bit

In addition, your system should have at least the following:

- 30 MB of RAM

- 300 MB of free disk space
- 50 MB of swap space

The size of the design that the Leda tools are checking determines the amount of additional main memory and disk space needed.

## Loading Leda

Leda is available via FTP and CD-ROM. In both cases, the Leda software is delivered in the form of a compressed tar file:

UNIX (Solaris/HP-UX/AIX):

```
leda_version_unix.tar.Z
```

Linux:

```
leda_version_linux.tar.Z
```

where, *version* is a 6-digit number (concatenation of year and month, for example: 200603) for the tool version and *architecture* is *gccsparcOS5/ linux/ amd64/ rs6000/ hp32/suse32/suse63*.

To load Leda via FTP, obtain the download instructions from Synopsys. To load Leda via CD-ROM, first mount the drive. [Table 2](#) lists some generic mount commands that may or may not be correct for your system, depending on its configuration. If necessary, check with your system administrator before using these commands.

**Table 2: Generic Mount Commands**

OS	Command
HP-UX	/etc/mount -r -F cdrfs device_name /cdrom
Solaris	Automatically mounted at /cdrom/cdrom0 if you are running vold daemon—if not, consult your system administrator.
Linux	/bin/mount -t iso9660 /dev/cdrom /mnt/cdrom
AIX	/etc/mount -v cdrfs -o ro

Once you have located the correct tar.Z file for your platform, copy that file to the installation directory (\$LEDA\_PATH).



### Attention

Make sure that you install the Leda software in a clean directory that does not contain previous versions of the Leda software.

**Caution**

---

Do not install Linux and UNIX versions of the Leda software in the same directory. They are not compatible. It is OK to have different UNIX platforms installed in the same directory (Solaris, HP-UX, and AIX).

---

## Global Installation (Required)

Before installing the software, set the `LEDA_PATH` environment variable to point to the root directory of the installation. You must have write permissions for the `$LEDA_PATH` directory. The Leda installation replaces all existing files in the target directory, so be sure to make copies of any customized data in the install tree before installing new versions of the software. Global installation involves the following tasks:

- [“Extracting Files” on page 12](#)
- [“Installing and Updating the Licenses” on page 15](#)
- [“License File Variable Settings” on page 16](#)
- [“Retrieving License Use Information” on page 16](#)

## Extracting Files

To extract the Leda files:

1. Uncompress the file:

```
% uncompress leda_version_platform.tar.Z
```

2. Untar the file using the below command:

```
% gtar xvf leda_version_platform.tar
```

After extraction, the contents of your installation directory (`$LEDA_PATH`) depend on whether you installed the UNIX or Linux version of the software:

- [UNIX Installation](#)
- [Linux Installation](#)

**Caution**

---

Do not overlay a UNIX installation on a Linux installation, or vice-versa. The UNIX and Linux versions of the Leda software are not compatible. They need to be installed in separate, clean directories that do not contain any previous versions of the Leda software.

---

**Attention**

---

Leda 4.2.x are the last releases that support RH7.2 version of Linux. Leda 4.3 and higher versions will not support RH7.2

---

## UNIX Installation

After extraction, the global installation is complete and the following directories are present in \$LEDA\_PATH for a UNIX installation:

- **auxx/**—contains Tcl libraries.
- **bin/**—contains a leda wrapper script that identifies what supported platform you are running on, and invokes the corresponding Leda executable software. The functions that you can access from the leda executable are determined by the licenses you purchased from Synopsys:
  - **leda**—Mixed-language VHDL and Verilog Specifier and Checker tool.
- **configurations/**—contains the four built-in rule configurations.
- **doc/**—contains Leda documentation that you can access directly from the Help pulldown menus in the Specifier and Checker tools. For an overview of the Leda documentation, read the *Leda Document Navigator* first.
- **icons/**—contains predefined icons that are used in the GUI display. If you want to use your own additional icons, put them in this directory too.
- **resources/**—Contains standard VHDL resource libraries STD and IEEE (versions compatible with VHDL 87 and VHDL 93), as well as the Synopsys library.
- **rules/**—contains rule source files for the prepackaged policies.
- **test/**—contains small projects used for testing the installation.
- **sparcOS5/**—contains utilities, flexlm, and bin subdirectories for Solaris. The bin directory contains platform-specific binaries for the Leda tools. The utilities and flexlm directories contain platform-specific utilities and licensing resources.
- **hpux11/**—contains utilities, flexlm, and bin subdirectories for HP-UX. The bin directory contains platform-specific binaries for the Leda tools. The utilities and flexlm directories contain platform-specific utilities and licensing resources.
- **rs6000/**—contains utilities, flexlm, and bin subdirectories for AIX. The bin directory contains platform-specific binaries for the Leda tools. The utilities and flexlm directories contain platform-specific utilities and licensing resources.
- **.leda\_config/**—contains the compiled database files for the prepackaged rules.

- **encoding/**—contains resource files for encoding Leda’s error messages in languages other than English.

## Linux Installation

After extraction, the global installation is complete and the following directories are present in \$LEDA\_PATH for a Linux installation:

- **auxx/**—contains Tcl libraries.
- **bin/**—contains a leda wrapper script that identifies what supported platform you are running on, and invokes the corresponding Leda executable software. The functions that you can access from the leda executable are determined by the licenses you purchased from Synopsys:
  - **leda**—Mixed-language VHDL and Verilog Specifier and Checker tool.
- **configurations/**—contains the four built-in rule configurations.
- **doc/**—contains Leda documentation that you can access directly from the Help pulldown menus in the Specifier and Checker tools. For an overview of the Leda documentation, read the *Leda Document Navigator* first.
- **icons/**—contains predefined icons that are used in the GUI display. If you want to use your own additional icons, put them in this directory too.
- **resources/**—Contains standard VHDL resource libraries STD and IEEE (versions compatible with VHDL 87 and VHDL 93), as well as the Synopsys library.
- **rules/**—contains rule source files for the prepackaged policies.
- **test/**—contains small projects used for testing the installation.
- **linux/**—contains utilities, flexlm, and bin subdirectories for Linux. The bin directory contains platform-specific binaries for the Leda tools. The utilities and flexlm directories contain platform-specific utilities and licensing resources.
- **.leda\_config/**—contains the compiled database files for the prepackaged rules.
- **encoding/**—contains resource files for encoding Leda’s error messages in languages other than English.

## Installing and Updating the Licenses

Leda is compatible with the Synopsys Common Licensing (SCL) program and uses the `snpslmd` daemon delivered with the SCL software. Before running the Leda software, you must install or update the license data file (`license.dat`) with the Leda licenses you purchased from Synopsys. Follow these steps:

1. If you maintain your Synopsys licenses on a common license server, you can update your `license.dat` file there to include your new Leda licenses and point your `LM_LICENSE_FILE` or `SNPSLMD_LICENSE_FILE` environment variable to the license server, as follows:

```
% setenv LM_LICENSE_FILE port@host
```

or

```
% setenv SNPSLMD_LICENSE_FILE port@host
```

where:

*port* is an available port on the license server, and

*host* is the host name

For example:

```
% setenv SNPSLMD_LICENSE_FILE 5300@cougar
```

2. If you maintain your licenses in an individual `license.dat` file, copy the license file obtained from Synopsys to `$LEDA_PATH/platform/flexlm/license.dat`

where:

*platform* is `sparcOS5`, `hpux11`, `rs6000`, or `linux`.

3. In the `license.dat` file `SERVER` line, change the *hostname* field to the name of the server where you are going to run the Leda license server. This must be the same machine used to generate the `hostid` that you provided to Synopsys to get your license keys. In the `VENDOR` line, add the full path to your Synopsys Common Licensing `snpslmd` daemon. Here is an example of a `license.dat` file:

```
SERVER hostname hostid
VENDOR snpslmd path_to_SCL_installation
INCREMENT leda-checker snpslmd version date N code
INCREMENT leda-specifier snpslmd version date N code
```

Each `INCREMENT` line contains the license information for activated parts of the software. The date field is the expiration date, the `N` field is the number of licenses, and the code field is an encryption code:

- m When you purchase the Specifier tool, you get a `leda-specifier` license `INCREMENT`. The Specifier license gives you access to both the Specifier and Checker tools.

- m When you purchase the standalone Checker tool, you get a leda-checker license INCREMENT. The Checker license gives you access to just the Checker tool.
4. If your site does not maintain a common license server and you are using an individual license.dat file, set the LM\_LICENSE\_FILE or SNPSLMD\_LICENSE\_FILE environment variable to point to \$LEDA\_PATH/platform/flexlm/license.dat:

```
% setenv LM_LICENSE_FILE $LEDA_PATH/platform/flexlm/license.dat
```

OR

```
% setenv SNPSLMD_LICENSE_FILE $LEDA_PATH/platform/flexlm/license.dat
```

5. If you are using your own license.dat file, activate the license server process as follows:

```
% cd $LEDA_PATH/platform/flexlm
% ./lmgrd -c license.dat >& /tmp/leda.license
```



### Note

If you are using the *port@host* method for your license server process, you don't need to complete Step 5. Your system administrator or licensing expert will probably activate the license server process.

## License File Variable Settings

Synopsys tools ignore the license file environment variable LM\_LICENSE\_FILE if SNPSLMD\_LICENSE\_FILE is set. The logic is as follows:

```
If SNPSLMD_LICENSE_FILE is set, then
    LM_LICENSE_FILE is ignored,
else
    LM_LICENSE_FILE value is used
```

## Retrieving License Use Information

The Leda software contains a utility in the \$LEDA\_PATH/platform/flexlm directory called lmutil that you can use to obtain information about license use. Through lmutil, the following utilities are available:

- **lmcksum**—prints license checksums
- **lmdiag**—diagnoses license checkout problems
- **lmdown**—shuts down license daemons on the license server node



- **lmhostid**—reports the hostid of the license server
- **lmreread**—causes the license daemon to reread the license file (useful for updating licenses)
- **lmstat**—helps monitor the current users of the different licenses

You can execute the lmutil utilities as follows:

```
% lmutil utility options
```

where *utility* is one of the above functions. The options available to these utilities are:

- *-c license\_file*
- *-verbose*

Examples:

```
% lmutil lmdown -c license.dat  
% lmutil lmstat -a
```

You can get more information about Flexlm from the Globetrotter Web site:

<http://www.globetrotter.com/TOC.htm>.

## Local Installation (Optional)

If you need special write access in order to create or modify Leda rules or resources, you can install the rules and resources locally. In this case, your license file must have a Specifier feature. Otherwise, the `setup_custom` script does not ask any questions about rule installation. Follow these steps:

1. Make sure `$LEDA_PATH` points to the root directory containing the Leda executables.
2. Set the `LEDA_CONFIG` environment variable to point to the configuration file present in the directory where you want to install a local copy of the rules (source and binaries). You must have write permissions for this directory. For example:

```
% setenv LEDA_CONFIG $HOME/Leda/config.tcl
```

If this directory does not exist, the `setup_custom` script asks you if you want to create it, and if the answer is Yes, tries to create it for you.

3. For VHDL only, if you want to install VHDL resource libraries locally, set the `LEDA_RESOURCES` environment variable to point to the directory where you want to install the resource libraries. You must have write permissions for this directory. For example:

```
% setenv LEDA_RESOURCES resources_directory
```

If this directory does not exist, the `setup_custom` script asks you if you want to create it, and if you answer Yes, tries to create it.

4. Once you have decided what you want to install locally, change the current directory to a directory that is not `$LEDA_PATH` (otherwise, you will overwrite the global installation). This can be a temporary directory or a specific one in which you want to create an example that you can use to test the installation.
5. Invoke the custom installation script as follows:

```
% $LEDA_PATH/utilities/setup_custom
```

The script prompts you for the items that it will install locally. It is advisable to answer Yes to all questions. You can change the options by answering No to the confirmation prompts. The script prints a summary of your choices and creates a `setup.log` file in the current working directory.

6. Once the correct configuration has been entered, the installation completes. To test your installation, follow the steps in [“Running the Post-Installation Test” on page 21](#).

## Configuring the Checker

This section explains how to set up the environment that controls the Checker configuration. You can set up the environment by setting environment variables in the UNIX shell or in an initialization file (`leda.ini`).

When you invoke the Checker, it searches for a configuration file (`leda.ini`) that can be either in the current working directory, `$HOME`, or `$LEDA_PATH`. This configuration file can contain settings for Checker environment variables. These settings overwrite any settings made in the UNIX shell. You define environment variables in the `leda.ini` configuration file as follows:

```
set variable value
```

Set your search path to include the Checker environment. For example:

```
% set path = ( $path $LEDA_PATH/bin )
```

## LEDA\_CONFIG

Set this environment variable to the configuration file that you want to use. Leda saves any subsequent changes that you make to the prepackaged rules in this same directory. Set this variable in the shell as follows:

```
% setenv LEDA_CONFIG configuration_file.tcl
```

or, in the `leda.ini` initialization file:

```
set LEDA_CONFIG configuration_file.tcl
```

If `LEDA_CONFIG` is defined before you invoke the Checker, the tool loads that configuration for prepackaged rules. Otherwise, the Checker uses the default configuration located in `$LEDA_PATH/.leda_config`.



### Caution

If you set `LEDA_CONFIG` to an empty directory before you invoke the tool, the Checker issues a warning message because it cannot find any policy configuration information. To solve this problem, exit the tool, unset the `LEDA_CONFIG` environment variable, and re-invoke the tool.

## LEDA\_RESOURCES (VHDL only)

The VHDL environment includes a directory called `$LEDA_PATH/resources` which contains standard VHDL resource libraries `STD` and `IEEE` (for VHDL 87 and VHDL 93), as well as the Synopsys resource library.

If you want to extend the `IEEE` library to include other vendor-specific packages or want to add other resource libraries to the global installation, you can edit or add resource libraries by following the steps described in the [Leda User Guide](#). This affects all users of the software at your site.

If you want to create local resource libraries for your own use, set the `LEDA_RESOURCES` environment variable to point to your local resource directory as follows:

```
% setenv LEDA_RESOURCES local_resource_directory
```

or, in the `leda.ini` initialization file:

```
set LEDA_RESOURCES local_resource_directory
```

If this variable is not defined, Leda uses the environment specified by `$LEDA_PATH/resources`.

## LEDA\_HTML\_DOC\_PATH and LEDA\_HTML\_USR\_PATH

When writing your own rules, you can add links to HTML-based help documents that you create so that other designers can access information explaining, for example, the rationale behind a certain rule. You can specify two different HTML documents for each rule you create, one for the original description of the rule and the other for revisions or application notes.

To indicate the location of these HTML documents, set the LEDA\_HTML\_DOC\_PATH and LEDA\_HTML\_USR\_PATH environment variables. The former points to the location of the original documentation and the latter points to the revision notes. Set these environment variables in the shell as follows:

```
% setenv LEDA_HTML_DOC_PATH "file:/standard_html_documentation_directory"  
% setenv LEDA_HTML_USR_PATH "file:/user_html_documentation_directory"
```

or, in the leda.ini initialization file:

```
set LEDA_HTML_DOC_PATH html_documentation_directory_1  
set LEDA_HTML_USR_PATH html_documentation_directory_2
```

If the LEDA\_HTML\_DOC\_PATH and LEDA\_HTML\_USR\_PATH variables are not defined, the directory used is \$LEDA\_PATH/doc/html.

## HTML\_NAVIGATOR

The Checker tool comes with HTML-based help files for the prepackaged rules that you can access from the Error Viewer in the Checker's main window. To indicate the location of the browser to be used to read the HTML documents, set the HTML\_NAVIGATOR environment variable with the full path to your browser as follows:

```
% setenv HTML_NAVIGATOR full_path_to_browser
```

or, in the leda.ini initialization file:

```
set HTML_NAVIGATOR full_path_to_browser
```

Example:

```
setenv HTML_NAVIGATOR /usr/local/bin/netscape
```

## LEDA\_READER

To access the PDF-based online documentation available from the Help pulldown menus in the Specifier and Checker tools, set the LEDA\_READER environment variable to point to the location of your PDF file reader. In most cases, the PDF file reader will be the Adobe Acrobat Reader tool, typically named acroread. Set this variable in the shell as follows:

```
% setenv LEDA_READER full_path_to_pdf_reader
```

or, in the leda.ini initialization file:

```
set LEDA_READER full_path_to_pdf_reader
```

Example:

```
% setenv LEDA_READER /usr/local/bin/acroread
```

## Running the Post-Installation Test

After you install and configure the software, you can run the demo design to test the installation. If the test/ directory was installed locally and LEDA\_CONFIG is defined, this is the example tested. Otherwise, the example in \$LEDA\_PATH/test/mixed (for mixed-language) is used. Whatever example you are using, you must have write permission to the directories in the environment.

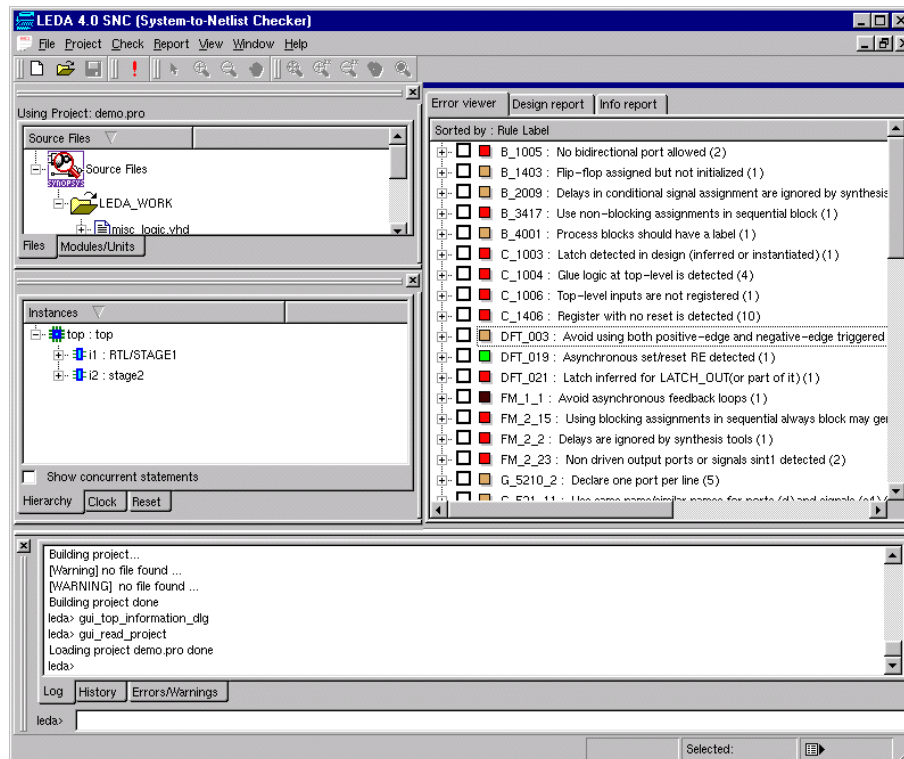
To run the post-installation test, follow these steps:

1. Invoke the Checker as follows:

```
% $LEDA_PATH/bin/leda &
```

This opens the Checker main window and a smaller Leda Project window that you can use to open an existing project or start a new one. Click on the Demo button to open the demo project. Then, click OK.

2. Leda runs the Checker automatically and displays results from the demo.pro project in the Error Viewer on the right side of the Checker's main window (see [Figure 1](#)). If the Checker runs without errors, your Leda installation is working.



**Figure 1: Checker Window after Demo Test**



# Index

- A**
- About this manual [5](#)
  - Acrobat [20](#)
  - AIX support [10](#)
- C**
- CD-ROM
    - loading [11](#)
  - Checker tool
    - configuration [18](#)
    - error report window [21](#)
    - options window [21](#)
  - Commands
    - gtar xvf [12](#)
    - leda [21](#)
    - mount [11](#)
    - uncompress [12](#)
  - Configuration
    - Checker tool [18](#)
  - Configuration file
    - leda.ini [19](#)
- D**
- Documentation conventions [6](#)
  - Drive, mounting [11](#)
- E**
- Environment variables
    - HTML\_NAVIGATOR [20](#)
    - LEDA\_CONFIG [18](#)
    - LEDA\_HTML\_DOC\_PATH [19, 20](#)
    - LEDA\_HTML\_USR\_PATH [19, 20](#)
    - LEDA\_PATH [18](#)
    - LEDA\_READER [20](#)
    - LEDA\_RESOURCES [19](#)
    - LM\_LICENSE\_FILE [16](#)
    - SNPSLMD\_LICENSE\_FILE [16](#)
  - Executables
    - leda [13, 14](#)
  - Extracting files [12](#)
- F**
- Files
    - configuration [18, 19](#)
    - extracting [12](#)
    - leda.ini [18, 19](#)
    - license.dat [15](#)
  - FTP
    - download instructions [11](#)
    - loading [11](#)
- G**
- Global installation [10, 17](#)
  - gtar xvf command [12](#)
  - Guide to Leda Documentation [13, 14](#)
- H**
- HP-UX support [10](#)
  - HTML\_NAVIGATOR environment variable [20](#)
- I**
- Installation
    - global [10, 17](#)
    - local [10, 17](#)
    - local rules [17](#)
- L**
- leda command [21](#)
  - Leda directory structure [12](#)
  - Leda Document Navigator [5](#)
  - leda.ini configuration file [18, 19](#)
  - LEDA\_CONFIG environment variable [18](#)
  - LEDA\_HTML\_DOC\_PATH environment variable [19](#)
  - LEDA\_HTML\_USR\_PATH environment variable [19, 20](#)

LEDA\_PATH environment variable [18](#)  
 LEDA\_READER environment variable [20](#)  
 LEDA\_RESOURCES environment  
   variable [19](#)  
 leda-executable [13](#), [14](#)  
 license.dat file [15](#)  
 Licensing [16](#)  
   Globetrotter Web site [17](#)  
   installing and updating [15](#)  
   license.dat file [15](#)  
   setting LM\_LICENSE\_FILE [16](#)  
   setting SNPSLMD\_LICENSE\_FILE [16](#)  
 Linux  
   Red Hat [10](#)  
 Linux support [10](#)  
 LM\_LICENSE\_FILE environment  
   variable [16](#)  
 Loading the Leda software [10](#)  
 Local installation [10](#), [17](#)

## N

Netscape [20](#)

## P

PDF reader [20](#)  
 Platform/OS Requirements [10](#)  
 Post-installation test  
   running [21](#)  
 Pre-installation requirements [9](#)

## R

Related documents [5](#)  
 Requirements  
   pre-installation [9](#)  
   system [10](#)

## S

Scripts  
   setup\_custom [18](#)  
 setup\_custom script [18](#)  
 Solaris support [10](#)  
 System requirements [10](#)

## T

Tools  
   mount [11](#)  
 Typographic and symbol conventions [6](#)

## U

uncompress command [12](#)  
 utilities [16](#)

## W

Web sites  
   Globetrotter [17](#)