cadence®

Virtuoso Studio Licensing and Configuration User Guide

Product Version IC23.1
September 2023

© 2023 Cadence Design Systems, Inc. All rights reserved.

Printed in the United States of America.

Cadence Design Systems, Inc. (Cadence), 2655 Seely Ave., San Jose, CA 95134, USA.

Open SystemC, Open SystemC Initiative, OSCI, SystemC, and SystemC Initiative are trademarks or registered trademarks of Open SystemC Initiative, Inc. in the United States and other countries and are used with permission.

Trademarks: Trademarks and service marks of Cadence Design Systems, Inc. contained in this document are attributed to Cadence with the appropriate symbol. For queries regarding Cadence's trademarks, contact the corporate legal department at the address shown above or call 800.862.4522. All other trademarks are the property of their respective holders.

Restricted Permission: This publication is protected by copyright law and international treaties and contains trade secrets and proprietary information owned by Cadence. Unauthorized reproduction or distribution of this publication, or any portion of it, may result in civil and criminal penalties. Except as specified in this permission statement, this publication may not be copied, reproduced, modified, published, uploaded, posted, transmitted, or distributed in any way, without prior written permission from Cadence. Unless otherwise agreed to by Cadence in writing, this statement grants Cadence customers permission to print one (1) hard copy of this publication subject to the following conditions:

1. The publication may be used only in accordance with a written agreement between Cadence and its customer.
2. The publication may not be modified in any way.
3. Any authorized copy of the publication or portion thereof must include all original copyright, trademark, and other proprietary notices and this permission statement.
4. The information contained in this document cannot be used in the development of like products or software, whether for internal or external use, and shall not be used for the benefit of any other party, whether or not for consideration.

Disclaimer: Information in this publication is subject to change without notice and does not represent a commitment on the part of Cadence. Except as may be explicitly set forth in such agreement, Cadence does not make, and expressly disclaims, any representations or warranties as to the completeness, accuracy or usefulness of the information contained in this document. Cadence does not warrant that use of such information will not infringe any third party rights, nor does Cadence assume any liability for damages or costs of any kind that may result from use of such information.

Cadence is committed to using respectful language in our code and communications. We are also active in the removal and replacement of inappropriate language from existing content. This product documentation may however contain material that is no longer considered appropriate but still reflects long-standing industry terminology. Such content will be addressed at a time when the related software can be updated without end-user impact.

Restricted Rights: Use, duplication, or disclosure by the Government is subject to restrictions as set forth in FAR52.227-14 and DFAR252.227-7013 et seq. or its successor

Contents

© 2023 All Rights Reserved.

© 2023 All Rights Reserved.

© 2023 All Rights Reserved.

© 2023 All Rights Reserved.

1

Configuring the Virtuoso Studio Design Environment

The *Virtuoso Studio Licensing and Configuration User Guide* describes how to configure the Virtuoso® Studio software as well as provides the following information about the environment:

- Executables needed to run the software - see and Setting the Installation Path for Cadence Tools for more information
- Operating system requirements - see Product Tier Features and Verifying Your System Configuration for more information
- Directory hierarchy for storing executables and data files - see Virtuoso Studio Design Environment Hierarchy and Setting the Installation Path for Cadence Tools for more information
- Key licensing characteristics - see Virtuoso Studio Design Environment Licensing Setup for more information

Prerequisites

This user guide is aimed at developers and designers of integrated circuits and assumes that you are familiar with:

- The Virtuoso Studio and application infrastructure mechanisms designed to support consistent operations between all Cadence® tools.
- The applications used to design and develop integrated circuits in the Virtuoso Studio, notably, the Virtuoso Layout Suite, and Virtuoso Schematic Editor.

The Virtuoso Studio design environment technology file.The following topics are discussed in this chapter:

- Virtuoso Studio
- Advanced Features and Token Licenses
- Methods to Track Token Licenses

© 2023 All Rights Reserved.

- Software Product License Management Form
- Product Tier Features
- Virtuoso Studio Design Environment Hierarchy
- Virtuoso Studio Design Environment Licensing Setup

Use the *Cadence Installation Guide* and *Cadence License Manager* to install the Virtuoso® Studio software and configure the licenses.

© 2023 All Rights Reserved.

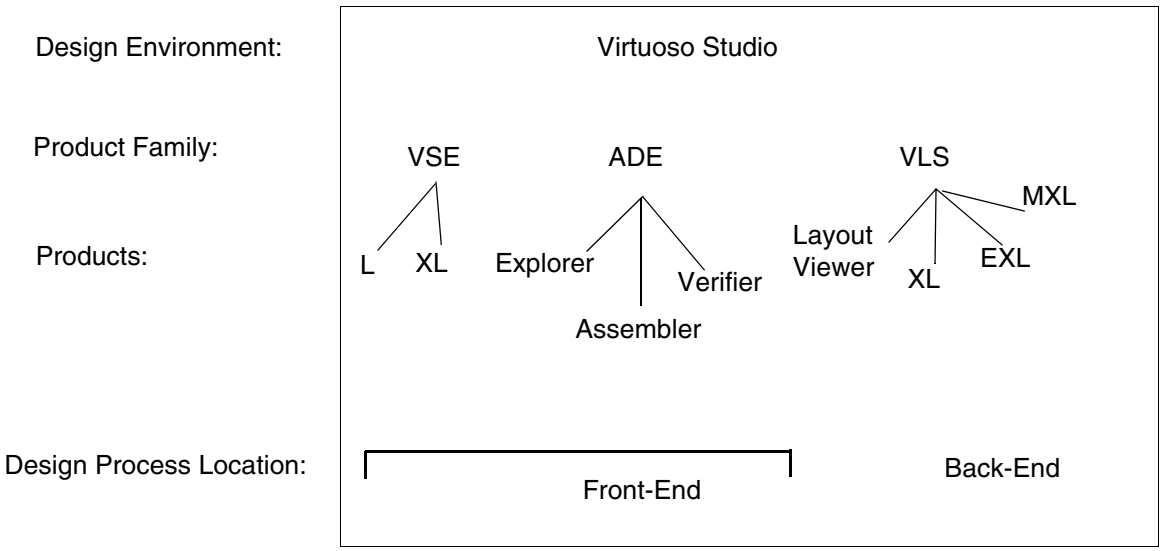
Virtuoso Studio

Virtuoso Studio provides a seamless transition from one Virtuoso Studio design environment to another with help of the common user interface (UI) and the automated license checkout process.

This platform provides a unified front-to-back design environment. The front-end design environment consists of Virtuoso Schematic Editor® (VSE) and ADE product families. The back-end consists of the Virtuoso Layout Suite® (VLS) product family.

Each of the product families contains applications in a tiered packaging structure, and conform to common and consistent licensing behavior and operation.

Overview of Virtuoso Studio Product Families



© 2023 All Rights Reserved.

Virtuoso Studio Licensing

When you purchase a Virtuoso product, you also purchase licenses that entitle you to use the applications that are part of that product. Most of these licenses are floating licenses that are available to anyone in the work group. They are checked out from a central license pool at the beginning of work and checked back in when work is done. A few licenses might be restricted to specific users or servers.

When you run Virtuoso software, you are more likely to use applications in combination with each other and the software checks out the necessary licenses automatically when you run the command to start an application. Alternatively, you can check licenses in and out from the Command Interpreter Window (CIW).

Application Levels of Virtuoso Studio License

Some Virtuoso applications are available at two or three different levels. The most basic read-only tier is Viewer, unassisted tier is L; the standard, assisted tier is XL; the advanced, automated tiers are EXL and MXL. Each higher tier provides additional features with more automated design assistance and you can also set the default application level for a cellview.

The following table shows the tiers for various Virtuoso applications:

Application	Family Tiers
Schematics	L, XL
Layout	XL, EXL, MXL
ADE	Explorer, Assembler, Verifier

The file `your_install_dir/share/license/products.dfII` contains product numbers and feature strings for Virtuoso software licenses.

You can check out and use applications of different levels together. The software always checks out the highest level license you need. If you descend in your design hierarchy, the software maintains the application level you are using.

If your attempt to check out a particular license level fails, the program asks whether you want to check out the next “higher-tiered” license. For example, if you cannot check out a license for Layout XL, the program asks whether you want to check out a higher-tiered license to run the Layout application. Your answer can be *Session*, *Skip*, *Always*, or *Never*. The *Session* and *Skip* options are stored in the memory. If you answer *Always*, the program automatically tries for a higher-tiered license without asking you again. If you answer *Never*, the program never tries for a higher-tiered license and never asks you again.The *Always* and *Never*

© 2023 All Rights Reserved.

options are automatically written in the .cdsenv file. When *Always* and *Never* options are used, current values for the licenses `VLSAdvOpt_UseNextLicense` (95511) and `VLSAdvStd_UseNextLicense` (95512) are used.

For L, XL, and EXL applications, such as Schematic, you can check out licenses one per user, host, display. You can open more than one session with a single license.

For GXL applications in the Layout applications, the software allocates tokens for the features you use.

Types of Licenses

- User, host, and display (UHD) licenses: The L, XL, EXL, and MXL product tiers have UHD licenses.
- Job (J)-based and token-based licenses: The Layout GXL product offers flexible licensing that can be used to invoke Layout XL and enable some advanced features in the Layout XL and EXL tiers. These are job-based (J) licenses which cannot be shared.

License Consumption in Product Tiers

Product Tier: L

The L product tier offers basic design creation and implementation capabilities.

Base Product Name	License Feature Name	License Type
Virtuoso Schematic Editor L	`Virtuoso_Schematic_Editor_L` (95100)	UHD

Important

The corresponding Virtuoso Layout L tier is no longer supported. . For more information, see *Virtuoso Layout Suite XL: Basic Editing User Guide* and *Virtuoso Layout Viewer User Guide*

© 2023 All Rights Reserved.

Product Tier: XL

The XL product tier contains IC design products that introduce new technologies and advancements in design automation. The XL tier offers tightened linkage between design and implementation phases in the areas of creation, analysis, implementation, and repair.

Base Product Name	License Feature Name	License Type
Virtuoso Layout Suite XL	`Virtuoso_Layout_Suite_XL` (95310)	UHD
Virtuoso Schematic Editor XL	`Virtuoso_Schematic_Editor_XL` (95115)	UHD

Product Tier: EXL

The EXL product tier offers full access to all the Layout XL functionality in addition to providing access to all the functionality that was accessible under Layout EAD in previous releases. The Layout EXL product tier is also the default application for advanced applications such as concurrent layout editing, design planning, congestion analysis and interactive simulation driven routing.

Layout EXL is also the minimum base application required for fully integrated access to layout tools and flows facilitating advanced layout design at 5nm process nodes and below.

For more information on enabling the EAD workspace and engines within Layout EXL, see *Virtuoso Electrically Aware Design Flow Guide*

Base Product Name	License Feature Name	License Type
Virtuoso Layout Suite EXL	`Virtuoso_Layout_Suite_EXL` (95800)	UHD

Only one base license per product family tier is checked out at a time. As a result:

- The VLS EXL tier does not require an VLS XL tier license. A higher checked out license allows you to run lower-tier applications. For example, you can run VLS XL and VLS EXL in the same session with only a VLS EXL license checked out. However, the higher checked out license will remain checked out until all VLS windows are closed.

  The following license is required in addition to the base license:

© 2023 All Rights Reserved.

- Token-based features in VLS GXL can be launched from the VLS XL or VLS EXL tier. (Tokens per Feature (VLS GXL))

Product Tier: MXL

The MXL product tier is the highest tier for layout application introduced in the Virtuoso Studio IC 23.1 release. It offers full access to all the Layout EXL functionality in addition to the integrated automated placement and routing solutions with unified interface to cater to all the design styles across both mature and advanced process nodes.

The Layout MXL tier is also the default cockpit for designing the integrated circuits in the context of the larger system-level design by providing technologies to address heterogeneous design, such as co-design and multi-fabric electromagnetic and thermal analysis.

Products that are not Tier-Based

Product Name	License Feature Name	License Type
Virtuoso Layout Viewer	`Virtuoso_Adv_Node_Framework` (95011)	UHD
Virtuoso ADE Explorer	`Virtuoso_ADE_Explorer` (95250)	UHD
Virtuoso ADE Assembler	`Virtuoso_ADE_Assembler` (95260)	UHD
Virtuoso ADE Verifier	`Virtuoso_ADE_Verifier` (95270)	UHD

For information on the specific features in each of the product tiers, see Software Product License Management Form: Diagnostics Tab.

Additional Feature Licenses

Some product features require you to check out additional licenses on top of the base product license for the product you are using.

Similarly, advanced node features require the `Virtuoso_Adv_Node_Opt_Layout` (Product Number `95511`), the `Virtuoso_Adv_Node_Opt_Lay_Std` (Product Number `95512`),or the `Virtuoso_Adv_Node_Layout_GAA` (Product Number 95513) license.

© 2023 All Rights Reserved.

Some of the additional licenses are not needed in the MXL tier as shown in table below.

Layout Tier	**VLS-XL (95310)**	**VLS-EXL (95800)**	**VLS-MXL (95810)**
Virtuoso MultiTech Option		+95022	✓
VLS GXL - Automation	+95323	+95323	
Virtuoso Photonics Basic optical editing and ORC		+ 95550	+ 95550
Virtuoso RF Multi-fabric RF and EM Analysis		+ 95560	+ 95560
Advanced Node > 10nm	+ 95512	+ 95512	+ 95512
Advanced Node <10nm	+ 95511	+ 95511	+ 95511
Advanced Node <5nm		+ 95511	+ 95511
Advanced Node GAA / 2nm		+ 95513	+ 95513

You can start applications in each Virtuoso product family using one of the following methods:

- *File – Open* and *File – New* from the CIW
- Library Manager
- *Launch* menu from the design windows
- *File – Open* and *File – New* from the design windows
- Hierarchy Editor
- Navigator Assistant
- History
- Bookmarks

Choosing Default Applications

Use one of the following methods to open a design in a specific product family tier and keep that setting as the default upon startup:

- *File – Open* and *File – New* from the CIW.

© 2023 All Rights Reserved.

- Library Manager.
- *File – Open* and *File – New* from the design windows
- *File – Set Default Application* from the design windows

To set the default application using any of the methods above:

1. Choose the desired application in the *Application* field.
2. Select the *Always use this application for this type of file* option.

   This field is mapped to the environment variables

   - `maskLayoutDefaultApp`
   - `schematicDefaultApp`
   - `schematicSymbolDefaultApp`

In the event that the default application is not a base application (L tier), only the license of the chosen application will be checked out, not the base application. Only one of the base licenses in a given family is ever checked out at any one time.

Setting the CDS_Netlisting_Mode Environment Variable

When working in ADE, you need to set the `CDS_Netlisting_Mode` SHELL environment variable to `Analog` using one of the following methods:

➡ Type the following at the command line prior to launching `virtuoso`

```
setenv CDS_Netlisting_Mode=Analog
```

➡ Type the following SKILL in the command interpreter window (CIW)

```
setShellEnvVar("CDS_Netlisting_Mode=Analog")
cdsSetNetlistMode()
t
cdsGetNetlistMode()
"Analog"
```

Design Framework II Licenses

Product licenses 111 and 95011 (`Virtuoso_Adv_Node_Framework`) are checked out at workbench start and while performing the following activities:

- Starting Virtuoso applications in either graph or non-graph mode

© 2023 All Rights Reserved.

- Launching routines that interact with design framework II® (DFII) data (for example, CDF, Pcells, and param)
- Accessing Cadence data using `itkDb`
- Launching open simulation system (OSS) netlisters

In the event of a DFII license failure at startup, the Virtuoso application exits and an error message is displayed in the CIW as well as a log file.

License Check Out/Check In Behavior

This topic discusses:

- License Usage When Using Interactive Commands
- License Usage When Using Non-Interactive Commands

License Usage When Using Interactive Commands

- **Product Tier**: **L**

  In the L product tier, if a schematic design is opened in read-only mode or changed to read-only mode by choosing *File – Make Read Only*, no license checkout is required.

  However, when a schematic design is opened for editing or when it is made editable by choosing *File – Make Editable* in the application window, the `Virtuoso_Schematic_Editor_L` license is checked out.

- **Product Tier**: **XL**

  In the XL product tier, the `Virtuoso_Layout_Suite_XL` license is checked out when a layout is opened. An additional `Virtuoso_Adv_Node_Opt_Layout`, `Virtuoso_Adv_Node_Opt_Lay_Std`, or `Virtuoso_Adv_Node_Layout_GAA` is checked out when the layout design contains Advanced Node features (). These licenses are checked in when all layout windows are closed, including all Layout Viewer, Layout XL, and Layout EXL windows.

- **Product Tier**: **EXL**

  In the EXL product tier, opening a design in Layout EXL checks out a `Virtuoso_Layout_Suite_EXL` license. An additional `Virtuoso_Adv_Node_Opt_Layout`, `Virtuoso_Adv_Node_Opt_Lay_Std`, or `Virtuoso_Adv_Node_Layout_GAA` is checked out when the layout design contains Advanced Node features ().

© 2023 All Rights Reserved.

- **Product Tier**: **MXL**

  In the MXL product tier, opening a design in Layout MXL checks out a `Virtuoso_Layout_Suite_MXL` license. An additional `Virtuoso_Adv_Node_Opt_Layout`, `Virtuoso_Adv_Node_Opt_Lay_Std`, or `Virtuoso_Adv_Node_Layout_GAA` is checked out when the layout design contains Advanced Node features ().

- **GXL Feature**

When you launch a GXL-based feature from the XL or EXL tier, the following token license activities occur:

- A token license is checked out during the execution of a command from a selected feature menu or toolbar.
- A token license is checked in automatically after completion of a command for some features.

The *Context* and *Window – Toolbar* menus do not affect the state of a token license. These menus are navigation aids and are not linked to license activity.

For all of the Virtuoso GXL features, with the exception of the Custom Digital Placer and Floorplanning, licenses are checked in after completion of the command. For the Custom Digital Placer and Floorplanning, the license is checked in only when the task is disabled or the XL or EXL session is exited.

Important

There is now an enforced restriction on license check-in that prevents license check in when an application or a GXL capability is still using the license.

License Usage When Using Non-Interactive Commands

Non-interactive, lengthy, and batch commands check out the license at the beginning of the command and automatically check in the license upon completion of the command activity. This model applies to automatic placers and routers.

© 2023 All Rights Reserved.

License Activity for Various Applications

License Requirements for Advanced Node and Advanced Methodologies Features

When you open or edit a design, the application checks if there is constraint or design element in use that requires the use of the layout advanced option license and the application requests for that license to be checked out. This license is referred to as the minimum data license. These licenses are listed in the table below from highest capability to the lowest capability.

License Feature (licName)	Product Number	Product Description
`Virtuoso_Adv_Node_Layout_GAA`	95513	Virtuoso Advanced Node Option for GAA
`Virtuoso_Adv_Node_Opt_Layout`	95511	Virtuoso Advanced Node Option for Layout
`Virtuoso_Adv_Node_Opt_Lay_Std`	95512	Virtuoso Advanced Node Option for Layout Standard

The following table lists the license requirements for different advanced node features:

Product Capability/Feature	License Required
Technology database contains features related to GAA	`Virtuoso_Adv_Node_Layout_GAA`

Technology database contains a layer with more than two masks	`Virtuoso_Adv_Node_Opt_Layout`
	or
	`Virtuoso_Adv_Node_Layout_GAA`

Layout XL cellview or the technology database contains WSP data. The license is not required in Layout EXL or higher tiers. See *Virtuoso Width Spacing Patterns User Guide*.	`Virtuoso_Adv_Node_Opt_Lay_Std`

© 2023 All Rights Reserved.

Product Capability/Feature	License Required
Cellview or technology database contains a constraint or constraint parameter that requires 95511 license. See Supported Constraints and Parameters: ICADVM20.1 Only-95511.	Virtuoso_Adv_Node_Opt_Layout or Virtuoso_Adv_Node_Layout_GAA
Technology database contains snap pattern definitions.	Virtuoso_Adv_Node_Opt_Lay_Std
Technology database contains a layer with 2 masks and no layer with more than 2 masks.	Virtuoso_Adv_Node_Opt_Lay_Std or Virtuoso_Adv_Node_Opt_Layout or Virtuoso_Adv_Node_Layout_GAA
Cellview or technology database contains a constraint or constraint parameter that requires 95512 license. See Supported Constraints and Parameters: Advanced Nodes Only.	Virtuoso_Adv_Node_Opt_Lay_Std or Virtuoso_Adv_Node_Opt_Layout or Virtuoso_Adv_Node_Layout_GAA
Advanced node features that are not listed in this table.	Virtuoso_Adv_Node_Opt_Lay_Std or Virtuoso_Adv_Node_Opt_Layout or Virtuoso_Adv_Node_Layout_GAA
When no advanced node data is present.	Neither Virtuoso_Adv_Node_Opt_Lay_Std nor Virtuoso_Adv_Node_Opt_Layout or Virtuoso_Adv_Node_Layout_GAA

© 2023 All Rights Reserved.

Advanced Node License Checkout Order

The checkout order of these licenses is controlled by the `VLSAdvOptLicenseCheckoutOrder` cadence environment variable. The default checkout order is:

```
"license" "VLSAdvOptLicenseCheckoutOrder" 'string "95512 95511 95513"
```

The license checkout order works when the license checkout request is being processed. This process always starts from the first license in the list until it finds a license that meets the checkout requirement and is available for checkout.

As you can use `VLSAdvOptLicenseCheckoutOrder` to change checkout order based on your requirements, Virtuoso takes the requesting license and goes over the checkout order value until it finds the first value that meets the requirements and can be checked out successfully.

License requested > License Available	`95512`	`95511`	`95513`
`95512`	Eligible for checkout	Not eligible	Not eligible
`95511`	Eligible for checkout	Eligible for checkout	Not eligible
`95513`	Eligible for checkout	Eligible for checkout	Eligible for checkout

You can also reset the checkout order using the controls in the Software Product License Management Form: Ordering Tab.

Virtuoso Visualization and Analysis XL and ADE Licenses

Virtuoso Visualization and Analysis tool runs only in the XL mode, which supports all the features of Virtuoso Visualization and Analysis L and XL from previous releases. It follows the license check out procedure given below.

- If Virtuoso Visualization and Analysis XL is opened from within ADE Explorer or ADE Assembler, it shares the same license tokens as that of the product.
- If Virtuoso Visualization and Analysis XL is opened in stand-alone mode or from Virtuoso, it can either check out the Virtuoso Visualization and Analysis XL license or a tier of Maestro license, depending upon the preferences you set in the

© 2023 All Rights Reserved.

VIVALicenseCheckoutOrder cdsenv variable. By default, this variable is set to VIVA, ADE, Verifier which results in the following license check out tasks being performed:

- Checks out the Virtuoso Visualization and Analysis XL license.
- If the checkout operation in the previous step fails, checks out a Maestro (Virtuoso ADE Explorer or Virtuoso ADE Assembler) license according to the checkout order specified in the checkoutOrder .cdsenv variable.

If this variable is set to ADE, VIVA, Verifier, the license check out tasks are performed in the following order:

- Checks out the license of ADE Explorer or ADE Assembler according to the checkout order specified in the checkoutOrder .cdsenv variable.
- If the checkout operation in the previous step fails, checks out the Virtuoso Visualization and Analysis XL license.

- When Virtuoso Visualization and Analysis XL is opened in stand-alone mode or from Virtuoso, the tool releases license when all the Virtuoso Visualization and Analysis XL windows are closed.
- When Virtuoso Visualization and Analysis XL is opened from within ADE and you close the ADE window, the Virtuoso Visualization and Analysis XL process holds the ADE license and continues to operate until all the main windows are closed.

License Usage in ADE Explorer and ADE Assembler

Depending on the run mode and feature being used ADE Explorer and ADE Assembler require different licenses. The following table describes the license consumption:

Table 1-1 License Checkout Behavior for ADE Explorer and ADE Assembler

Action	**License Checkout Preference Order (depending on the availability)**
Open ADE Explorer	Virtuoso_ADE_Explorer Virtuoso_ADE_Assembler
Open ADE Assembler	Virtuoso_ADE_Assembler
Switch from ADE Explorer to ADE Assembler	Check in Virtuoso_ADE_Explorer license and check out Virtuoso_ADE_Assembler license

© 2023 All Rights Reserved.

Action	License Checkout Preference Order (depending on the availability)
Switch from ADE Assembler to ADE Explorer	Use only `Virtuoso_ADE_Assembler` license

In addition to the Virtuoso ADE Assembler license, you would require the `Virtuoso_Variation_Option` (95265) license for the following advanced options in Virtuoso ADE Assembler:

- High-sigma MC analysis to identify the worst-case behavior.
- Sensitivity accuracy and mismatch contribution.
- Assisted Yield Improvement flow.
- Advanced Optimization algorithms.

When running a large number of parallel jobs from ADE Explorer or Assembler, the Virtuoso Simulation Expansion Option (95290) license is required for every increment of 400 jobs. Also, Virtuoso ADE Simulation Manager (95237) license is required to enable maximum simulation throughput and responsiveness in ADE Assembler.

License Checkout When Running the Virtuoso Schematic and Verilog Driven Mixed-Signal Flow

To run the Virtuoso Schematic and Verilog Driven Mixed-Signal Flow, you must set the `CDS_ENABLE_VMS` environment variable before starting Virtuoso.

License Checkout While Running the Electrically Aware Design (EAD) Flow

While running simulations in Virtuoso ADE Assembler to generate electrical data and save datasets for the EAD flow, in addition to the base license for Virtuoso ADE Assembler, the EAD flow requires either of the following licenses:

- `Virtuoso_Layout_Suite_EXL` (Product Number 95800)
- `Virtuoso_Variation_Analysis_Op` (Product Number: 95510)

  Note: If 95800 is already checked out, 95510 is not used.

You can change the default license checkout order using the `EADdatasetCheckoutOrder` environment variable.

© 2023 All Rights Reserved.

License Checkout for Virtuoso Photonics Platform

To enable the photonics flows, you can set either the `Virtuoso_Photonics_Platform` shell or the `Virtuoso_Photonics_Option` environment variable.

After setting the `Virtuoso_Photonics_Platform` variable, when you start Virtuoso, an attempt is made to check out and initialize the `Virtuoso_Photonics_Platform` license (Product Number 95551). If this license is available, you will not need separate licenses for Virtuoso CurvyCore technology, Virtuoso Schematic Editor, Virtuoso Layout Suite, and Analog Design Environment features.

Alternatively, you can also set the `Virtuoso_Photonics_Option` environment variable to enable the photonics features. After setting this environment variable, when you start any Virtuoso application, such as Virtuoso, stream, itkDB-based binaries, an attempt is made to check out and initialize the `Virtuoso_Photonics_Option` license (Product Number 95550). Subsequently, as you open editing cockpits, relevant product licenses will be checked out (for example, VLS-EXL for layout editing), as needed.

In both the cases, if the specified license is not found, the launch of the executable fails.

To enable Virtuoso CurvyCore technology, you would need the `Virtuoso_MultiTech_Framework` license in addition to the `Virtuoso_Photonics_Option` license. However, in case of `Virtuoso_Photonics_Platform` license, Virtuoso CurvyCore technology is enabled automatically.

Note: You can use the `dbIsPhotonicsEnabled()` API to check if the photonics features have been enabled.

License Checkout for Virtuoso MultiTech Framework

To enable Virtuoso MultiTech Framework, you need to set the shell environment variables `Virtuoso_MultiTech`.

After setting one of the above shell environment variables, when you start Virtuoso, the following licenses are checked out:

- Cadence Design Framework II (Product Number 111)
- `Virtuoso_Adv_Node_Framework` (Product Number 95011)
- `Virtuoso_MultiTech_Framework` (Product Number 95022)

© 2023 All Rights Reserved.

License Check out for Virtuoso RF IC Design Solution

- `Virtuoso_Adv_Node_Framework`
- `Virtuoso_Schematic_Editor_XL`
- `Virtuoso_Layout_Suite_EXL`

One of the following Cadence EM Solvers:

- `EMX`
- `Clarity_3DSolver`

Virtuoso Layout Suite EXL includes:

- `Virtuoso_Adv_Node_Framework` license feature.
- The Electromagnetic Assistant and the interface to EM solvers.

License Check out for Virtuoso RF System Design Solution - Module Layout in Virtuoso

- `Virtuoso_MultiTech_Framework`
- `Virtuoso_Schematic_Editor_XL`
- `Virtuoso_Layout_Suite_EXL`
- `Clarity_3DSolver`

Virtuoso Layout Suite EXL includes the Electromagnetic Assistant and the interface to EM solvers.

You must set the following shell environment variable before you launch Virtuoso:

- `Virtuoso_MultiTech`

This environment variable is not needed in Layout MXL.

License Check out for Virtuoso RF System Design Solution - Module Layout in Allegro

- `Virtuoso_MultiTech_Framework`
- `Virtuoso_Schematic_Editor_XL`
- Allegro Package Designer Plus

© 2023 All Rights Reserved.

- SiP Layout Option
- `Clarity_3DSolver`

You must set the following shell environment variable before you launch Virtuoso:

- `Virtuoso_MultiTech`

License Checkout for Virtuoso Power Manager

- `Virtuoso_Schematic_Editor_XL`
- `VIRTUOSO_POWER_MANAGER`

License Checkout When Running PVS-CV in Virtuoso

When running Physical Verification System (PVS) Constraint Validation (CV) in Virtuoso, the following license is checked out in addition to the base product licenses, which are checked out depending on the requirement of other features being used:

- `Phys_Ver_Sys_Const_Validator` (Product Number 96300)

Important

Use PVE12.1.1 version or later for running the PVS-CV feature.

License Checkout When Running Virtuoso IPVS

Virtuoso Integrated Physical Verification System (IPVS) requires the following licenses:

Basic Licenses:

- `Virtuoso_DRC_Opt`: Virtuoso Integrated Physical Verification System Option for Virtuoso Layout Suite license (Product Number: 96400)
- `Phys_Ver_Sys_Results_Mgr`: Cadence Physical Verification System Results Manager (Product Number: 96240)

Advanced Licenses:

- `Virtuoso_IPVS_Adv_Ana_Opt`: Virtuoso Integrated Physical Verification System Advanced Analysis Option (Product Number:96310)

Note: For more information, see *Licenses Requirements and Behaviors* in the *Virtuoso IPVS User Guide.*

© 2023 All Rights Reserved.

License Checkout for Running Third-party Simulators

For running third-party (non-MMSIM) simulators in ADE product family (Explorer/Assembler) the Virtuoso® Analog Oasis Run-Time Option license (`OASIS_Simulation_Interface`, Product Number 32100) is required. Prior to usage, the simulator must be authorized for integration into the simulation environment. To integrate a commercially available simulator contact Cadence Connections (connections@cadence.com) and to integrate a proprietary simulator, contact Cadence Customer Support.

Once the integration is complete, the Oasis Run-Time Option license will be checked out and held during netlisting and simulation. The license is checked back in when the netlisting and simulation run is complete.

License Checkout When Running Express Pcell in Virtuoso

The Express Pcell capabilities are available through Cadence Design Framework II (Product Number 111) without a need to check out the VLS XL license.

Abstract Generator License Usage

Abstract Generator requires the `Virtuoso_Layout_Suite_XL`, `Virtuoso_Layout_Suite_EXL`, or `Virtuoso_Layout_Suite_MXL` and `Virtuoso_Adv_Node_Opt_Layout`, `Virtuoso_Adv_Node_Opt_Lay_Std`, or `Virtuoso_Adv_Node_Layout_GAA` licenses.

License Checkout When Running Design Planning and Analysis

Design Planning and Analysis requires `Virtuoso_Layout_Suite_EXL` and 12 GXL Tokens or `Virtuoso_Layout_Suite_MXL`.

Pcells and License Usage

Pcell Interoperability and Performance: The actions of saving, editing, or opening a database using Core Cached Pcells capability, from either the GUI or SKILL and from any application, causes a `Virtuoso_Layout_Suite_XL` license to be checked out.

Process Rule Editor: The use of the Process Rule Editor, or associated SKILL function, to save, edit, or open a techDB or design object requires a `Virtuoso_Layout_Suite_XL` or `Virtuoso_Schematic_Editor_XL` license to be checked out.

© 2023 All Rights Reserved.

Summary of License Checkout Behavior

The following table summarizes the license check-out behavior for different product features:

Table 1-2 Summary of License Check-Out Behavior

Product Capability/Feature	Base Product License Required	Additional License Required	Close Application Window[1]
VSE L (Read only mode)	None	None	NA
VSE L (Edit mode)	`Virtuoso_Schematic_Editor_L`	None	License checked in
VSE XL	`Virtuoso_Schematic_Editor_XL`	None	License checked in
ADE Explorer	`Virtuoso_ADE_Explorer` or `Virtuoso_ADE_Assembler`	None	License checked in
ADE Assembler	`Virtuoso_ADE_Assembler`	None	License checked in
ADE Verifier	`Virtuoso_ADE_Verifier`	The implementations whose simulations are started from Verifier require their appropriate licenses.	License checked in
Command-Line IP Selector	AMS_environment (Product Number 70000)	None	License checked in
VLS XL	`Virtuoso_Layout_Suite_XL` or `Virtuoso_Layout_Suite_EXL` or `Virtuoso_Layout_Suite_MXL`	`Virtuoso_Adv_Node_Opt_Layout` or `Virtuoso_Adv_Node_Opt_Lay_Std` or `Virtuoso_Adv_Node_Layout_GAA`	License checked in

© 2023 All Rights Reserved.

Product Capability/Feature	Base Product License Required	Additional License Required	Close Application Window[1]

VLS EXL	`Virtuoso_Layout_Suite_EXL`
	or
	`Virtuoso_Layout_Suite_MXL`
		`Virtuoso_Adv_Node_Opt_Layout`
		or
		`Virtuoso_Adv_Node_Opt_Lay_Std`
		or
		`Virtuoso_Adv_Node_Layout_GAA`
			License checked in

VLS MXL	`Virtuoso_Layout_Suite_MXL`	`Virtuoso_Adv_Node_Opt_Layout`
		or
		`Virtuoso_Adv_Node_Opt_Lay_Std`
		or
		`Virtuoso_Adv_Node_Layout_GAA`
			License checked in

[1] Close all windows running this application except the CIW.

* The `Virtuoso_Adv_Node_Opt_Layout`, `Virtuoso_Adv_Node_Opt_Lay_Std`, or `Virtuoso_Adv_Node_Layout_GAA` license is checked in when the last layout window is closed.

© 2023 All Rights Reserved.

Setting License Check Environment Variables

There are a range of license check environment variables that can be used for detailed license checkout reporting. For example, you can set an environment variable to report whether a license check is taking too long or if a particular license or token has been checked out.

These license environment variables, as shown in an example for each below, can be set in one of two ways:

1. Using `setenv` before running Virtuoso.
2. Using the SKILL function `setShellEnvVar()` during a Virtuoso session.

- `CDS_LIC_PRINT_FILTER`

  This environment variable can contain a string with the names of license features and token capabilities, for detailed license checkout reporting using the format:

  "`Feature1:Token1,Token2,...:TokenN,Feature2, ...,FeatureM`"

  If `CDS_LIC_PRINT_FILTER` is not defined, the license feature “111” will be the default.

  You can obtain a list of all registered token capabilities by examining the content of the Software Product License Management Form: Token Usage Tab.

  For example:

  ```
  setenv CDS_LIC_PRINT_FILTER "111, Virtuoso_Layout_Suite_XL,
  Virtuoso_Layout_Suite_GXL:VLS_GXL"
  setShellEnvVar("CDS_LIC_PRINT_FILTER=111, Virtuoso_Layout_Suite_XL,
  Virtuoso_Layout_Suite_GXL:VLS_GXL")
  ```

  You can also use a number of aliases with the `CDS_LIC_PRINT_FILTER` environment variable to cover multiple feature settings:

Environment Variable Value Alias	**License Tool Interprets** `CDS_LIC_PRINT_FILTER` **for**
`all`	All tiered licenses, including all token capabilities and license “111”.
`vlsall`	All licenses related to the Virtuoso Layout Suite family.
`adeall`	All licenses related to the Analog Design Environment family.
`vseall`	All licenses related to the Virtuoso Schematic Editor family.

© 2023 All Rights Reserved.

When setting the CDS_LIC_PRINT_FILTER environment variable value, you can also combine application names and aliases, for example:

```
setenv CDS_LIC_PRINT_FILTER
"Virtuoso_Layout_Suite_XL,vseall,adeall":virtuoso
setenv CDS_LIC_PRINT_FILTER "vlsall,vseall,adeall,111":virtuoso
```

Note: If you set "Virtuoso_Layout_Suite_GXL:all", the license tool will interpret CDS_LIC_PRINT_FILTER as containing all token capabilities for the token license "Virtuoso_Layout_Suite_GXL".

- CDS_LIC_PRINT_TIME

  This boolean environment variable can be set to output the time (in milliseconds) to the CIW that is taken during a license check-out process. The values that can be set for CDS_LIC_PRINT_TIME are 0 (do not print time) and 1 (print time), the default being 1.

  For example:

  ```
  setenv CDS_LIC_PRINT_TIME 0
  or
  setShellEnvVar("CDS_LIC_PRINT_TIME=0")
  ```

- CDS_LIC_PRINT_ALWAYS

  Note: This boolean environment variable can be set to output messages for a particular license feature each time that license feature is checked out. The values that can be set for CDS_LIC_PRINT_ALWAYS are 0 (do not print each time) and 1 (print each time), the default being 0.

  Note: If CDS_LIC_PRINT_ALWAYS is not defined, or its value is 0, the message will be output only one time to the CIW.

  For example:

  ```
  setenv CDS_LIC_PRINT_ALWAYS 1
  or
  setShellEnvVar("CDS_LIC_PRINT_ALWAYS=1")
  ```

- CDS_LIC_TIMER_INTERVAL

  This environment variable is used to set the number of seconds required to pass, for a license search, before a timeout occurs. If the license check time exceeds the value of CDS_LIC_TIMER_INTERVAL, a message will be output to the CIW informing you that the check time has been exceeded. The default value is 15 seconds.

  For example:

  ```
  setenv CDS_LIC_TIMER_INTERVAL 10
  or
  setShellEnvVar("CDS_LIC_TIMER_INTERVAL=10")
  ```


© 2023 All Rights Reserved.

- `CDS_LIC_PRINT_TYPE`

  This environment variable can be set to output all license call related messages (information and warning) to the CIW, or only the warning messages. If the value of `CDS_LIC_PRINT_TYPE` is set to "`warn`", then only warning messages will be output to the CIW. If set to "`all`", both information and warning messages will be output. The default is "`warn`".

  For example:

  ```
  setenv CDS_LIC_PRINT_TYPE "all"
  or
  setShellEnvVar("CDS_LIC_PRINT_TYPE=all")
  ```

- `CDS_LIC_TIMEOUT_DIALOG`

  This environment variable is used to display the timeout dialog when the license checkout time of a license feature, as defined by `CDS_LIC_PRINT_FILTER`, exceeds the timer interval specified by `CDS_LIC_TIMER_INTERVAL`. The value of `CDS_LIC_TIMEOUT_DIALOG` can be either "`1`" (which will display the dialog) or "`0`" (which will not display the dialog). The default value is "`0`".

  Note: For the initial 111 license, the timeout dialog will be displayed even when there is no license 111 in `CDS_LIC_PRINT_FILTER` and irrespective of `CDS_LIC_TIMEOUT_DIALOG` being set.

- `CIC_ENABLE_LIC_PERF`

  This environment variable tests the license checkout performance of license features. If the license checkout time exceeds the specified threshold, a warning message is printed to the CIW.

  For example:

  ```
  setenv CIC_ENABLE_LIC_PERF "limit=0.08 pings=4 interval=0.5"
  ```

  Where,

  `limit = [time]` is the threshold value for checkout time (in seconds).

  `pings = [count]` is the number of pings sent to the license server. When this option is set, a specified number of pings will be sent to each license server (defined in CDS_LIC_FILE), when the license request time exceeds the threshold value. The default is `0`, which means no pings are sent to the license server.

  `interval = [time]` defines the time interval (in seconds) between pings.

  The results from the ping commands are printed to the CIW.

© 2023 All Rights Reserved.

Advanced Features and Token Licenses

The advanced technology features in the XL and EXL tiers are accessed by tokens or multi-feature licenses which are defined by a multi-feature license model.

Note: See also Methods to Track Token Licenses.

A token is one copy of a multi-feature license. Each feature requires a specific number of tokens. GXL multi-feature licenses are Job-based (J) licenses which are not shared between processes.

GXL Multi-Feature License Tokens

A specific number of tokens is required to access advanced features in `Virtuoso_Layout_Suite_XL` and `Virtuoso_Layout_Suite_EXL`. Tokens are not needed when same features are accessed from the Layout MXL tier.

If all of your license tokens are currently in use, or you have insufficient tokens to access a particular feature, additional tokens can be obtained by contacting your local Cadence representative. The following table lists the tokens needed for the VLS GXL multi-feature licenses:

Table 1-3 Tokens per Feature (VLS GXL)

Features	VLS_XL	VLS_EXL	Version
Analog_Auto_Placer	8	8	23.1
Congestion Analysis		12	23.1
Concurrent Layout Editing	4	0	23.1
Design_Planning_Analysis		12	23.1
Digital_Auto_Placer	2	2	23.1
Digital_Auto_Placer_Adv		24	23.1
Floorplanning	4	4	23.1
Integrated_Short_Locator	3	3	23.1
Module_Generator	2	0	23.1
Slotting	4	0	23.1
Space_based_Router	12	12	23.1

© 2023 All Rights Reserved.

Features	VLS_XL	VLS_EXL	Version
Symbolic_Placement_Devices	2	2	23.1
VPLGen	1		23.1

Important Points to Note

- All of the features listed in Table 1-3 can be started in the XL, EXL, or MXL window.
- VPLGen: Using any of the Virtuoso Parameterized Layout Generator commands or Virtuoso Layout Generator commands to create, edit, or update a VPLGen or VLGen causes one VLS GXL token to be checked out. This GXL token remains checked out for the duration of the VLS XL or EXL session.
- Slotting does not require any GXL tokens in the Layout EXL and Layout MXL tier.

Virtuoso Space-based Router Tokens

Some Virtuoso Space-based Router features, such as viewing, navigation, and DRC checking, can be run from Virtuoso Layout Suite XL or EXL without additional licensing. Space_based Router tokens are required to perform routing and other tasks that modify the routing and routing environment, based on the number of nets in the design. When you invoke the first command that requires Space_based Router tokens, the tokens are checked out and are not released until you exit Virtuoso Space-based Router.

Some Virtuoso Space-based Router commands can use multiple processors in one workstation to accelerate processing. Additional Space_based Router tokens are required for the additional processors. The below table shows the number of additional tokens that you will need, based on the number of processors, or threads, that you want to use. License tokens can be reserved globally for use by all commands in a session or can be checked out as needed for a command. Tokens Per Feature: VSR for Automatic Routing (Space_based_Router)

Number of Nets	Number of Processors	Number of Tokens
0 to 1	N/A	VLS XL license required
2 to 9999	1	12
2 to 9999	2 to 4	24 (+12)
2 to 9999	5 to 8	36 (+12)
2 to 9999	9 to 12	48 (+12)

© 2023 All Rights Reserved.

2 to 9999	each additional 4 processors (over 4 processors)	+ 12 tokens
10,000 to 249,999	1	24
10,000 to 249,999	2 to 4	48 (+24)
10,000 to 249,999	5 to 8	72 (+24)
10,000 to 249,999	9 to 12	96 (+24)
10,000 to 249,999	each additional 4 processors (over 4 processors)	+ 24 tokens
>= 250,000	1	48
>= 250,000	2 to 4	72 (+24)
>= 250,000	5 to 8	96 (+24)
>= 250,000	9 to 12	120 (+24)
>= 250,000	each additional 4 processors (over 4 processors)	+ 24 tokens

For example, if your design has 100,000 nets and you want to run Virtuoso Space-based Router using six (6) processors, you will need 48 Space_based Router tokens, 24 for the net count plus 24 for the thread count.

© 2023 All Rights Reserved.

Methods to Track Token Licenses

The following methods can be used to track token license activity.

- lmstat Command
- Software Product License Management Form

lmstat Command

The `lmstat` command can be used to track token license activity from the command line. It reports activity for one license server only. Use the following arguments with the `lmstat` command.

Option	Description
`-a`	Display everything
`-c license_file`	Use "license_file" as license file
`-f [feature_name]`	List usage info about specified (or all) features
`-i [feature_name]`	List info about specified (or all) feature(s) from the increment line in the license file
`-S [DAEMON]`	Display all users of DAEMON licenses
`-s [server_name]`	Display status of all license files on server node(s)
`-t timeout_value`	Set connection timeout to "timeout_value"
`-v`	Display FLEXlm version, revision, and patch
`-old`	Allow communications with an old server that uses communications version 1.2 or earlier
`-help`	Print this message

© 2023 All Rights Reserved.

Example

1. Type the following in a shell window:

   ```
   lmstat -c <license.file> -f "<feature.name>"
   ```

   For example,

   ```
   lmstat -c sj234flex4 -f "Virtuoso_Layout_Suite_GXL"
   ```

2. Review the output.

   In this example, `designer1` on host `lne2` has the `VLS_GXL` license checked out (4 tokens).

```
Users of Virtuoso_Layout_Suite_GXL:
 (Total of 400 licenses issued;  Total of 4 licenses in use)


  "Virtuoso_Layout_Suite_GXL" v6.100, vendor: cdslmd floating license


designer1 lne2 VLS_GXL (v6.100) (server 54), start Tue 4/17 2:40, 4 licenses
```

   For more information about the `lmstat` command, see the *Cadence License Manager*.

   Note: The `tokenPrintCapabilities` SKILL function that printed the details of registered token licenses, including the number of tokens required for each capability and whether or not it is in use, no longer prints the token details. It will be deprecated in a future release. For more information on `tokenPrintCapabilities`, refer to the *Virtuoso Studio Design Environment SKILL Reference*.

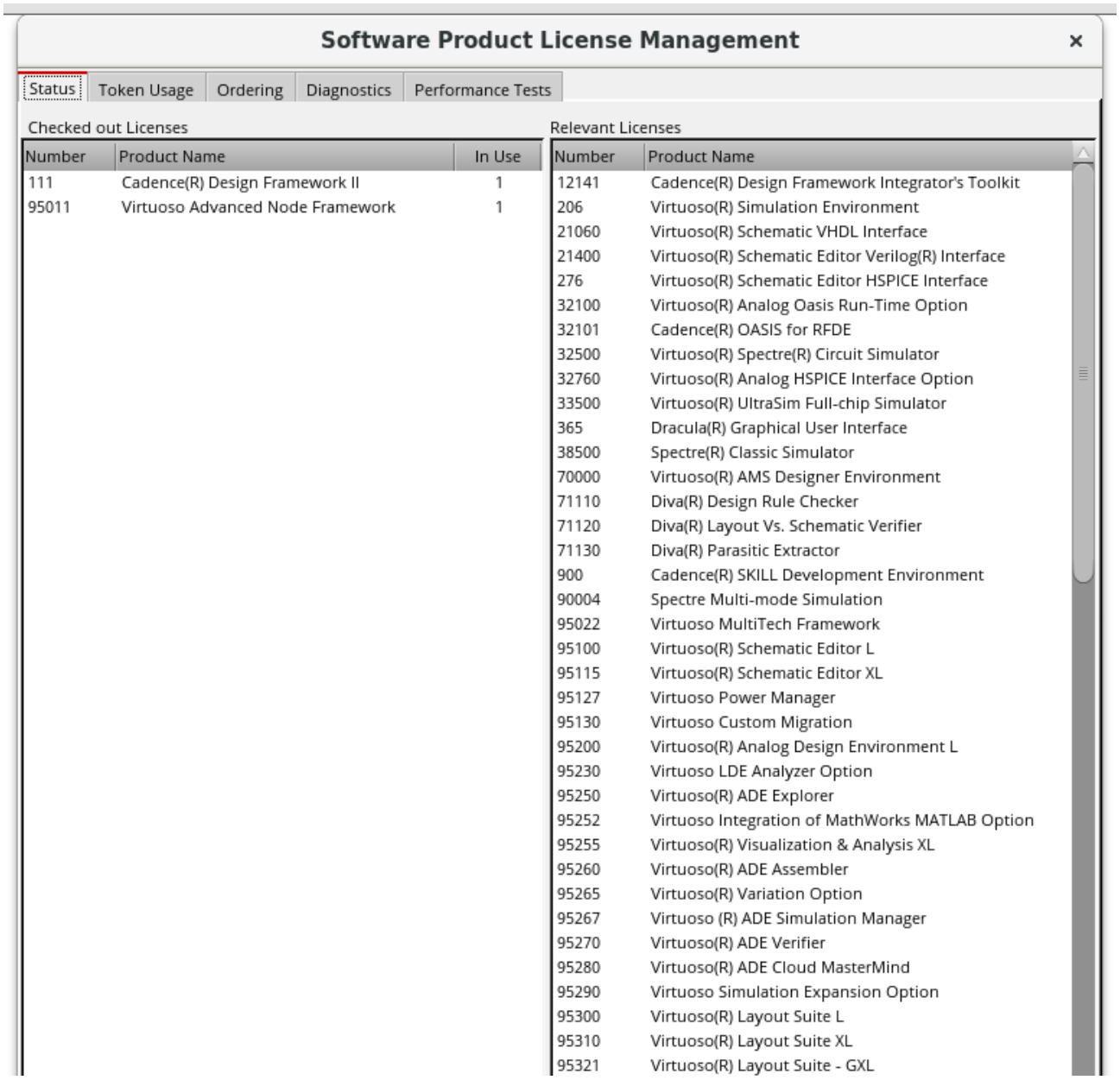
© 2023 All Rights Reserved.

Software Product License Management Form

You can use the Software Product License Management form to manage the licensing activity.

To open the form:

➡ Select *Options – License* in the CIW. The Software Product License Management form is displayed. In this form, you can get all the Cadence-product license related information.



© 2023 All Rights Reserved.

The form contains the following tabs to handle the licensing process:

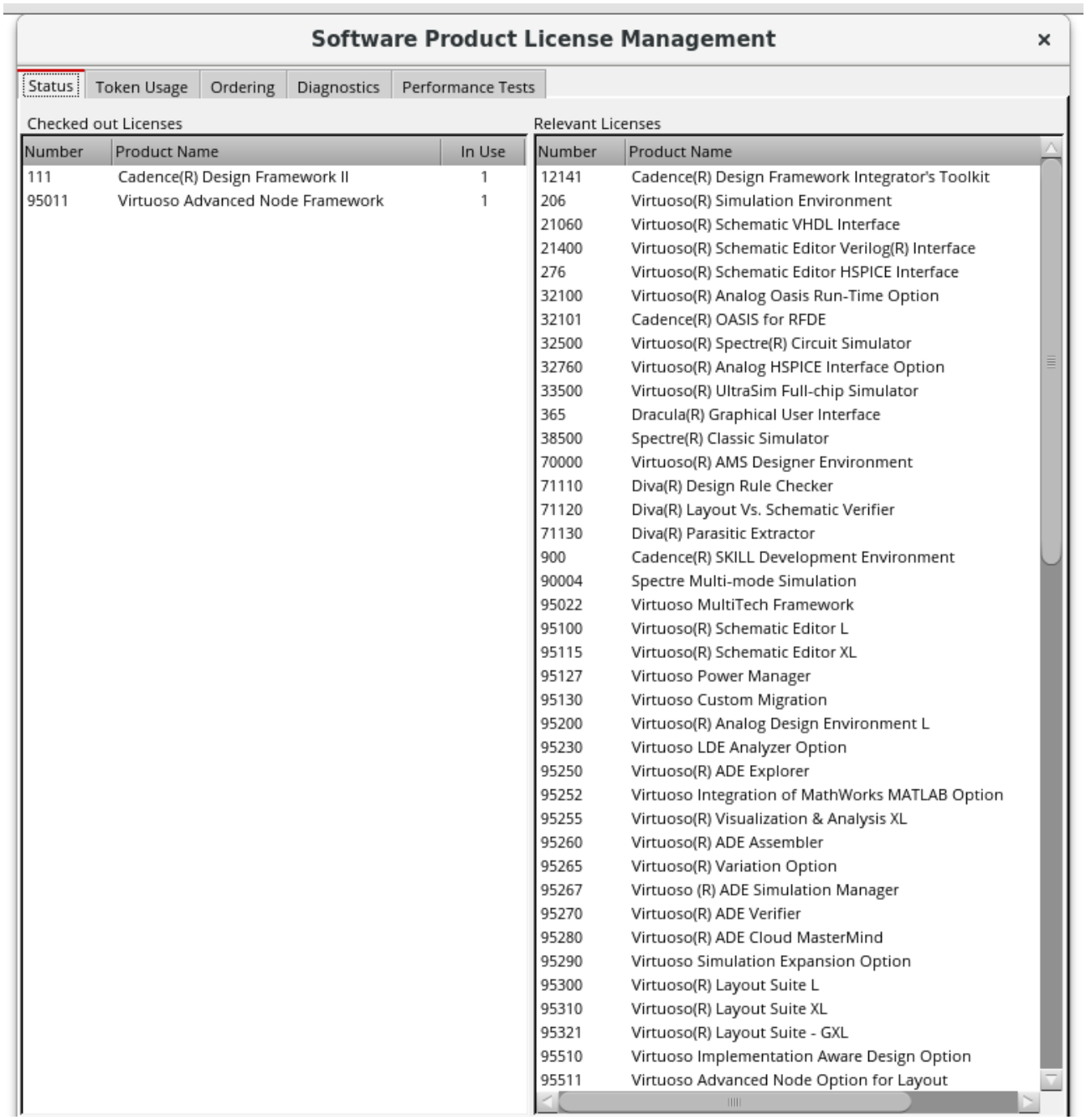
- **Status**: Displays the license status in two panes – *Checked Out Licenses* and *Relevant Licenses*. For more information, see the Software Product License Management Form: Status Tab.
- **Ordering**: Contains the options related to license ordering. For more information, see the Software Product License Management Form: Ordering Tab.
- **Diagnostics**: Contains the options for configuring the license diagnostic settings. For more information, see the Software Product License Management Form: Diagnostics Tab.
- **Performance Tests**: Contains the options to run a diagnostic test that determines the license-checkout performance on a feature-by-feature basis and across each license server. The tests can be looped and repeated over a period of time to calculate the average, median, max, and min times for each checkout. For more information, see the Software Product License Management Form: Performance Tests Tab.

© 2023 All Rights Reserved.

Software Product License Management Form: Status Tab

To open the Software Product License Management form - Status tab

➡ Select *Options – License* in the CIW. The Software Product License Management form is displayed with the *Status* tab selected by default.



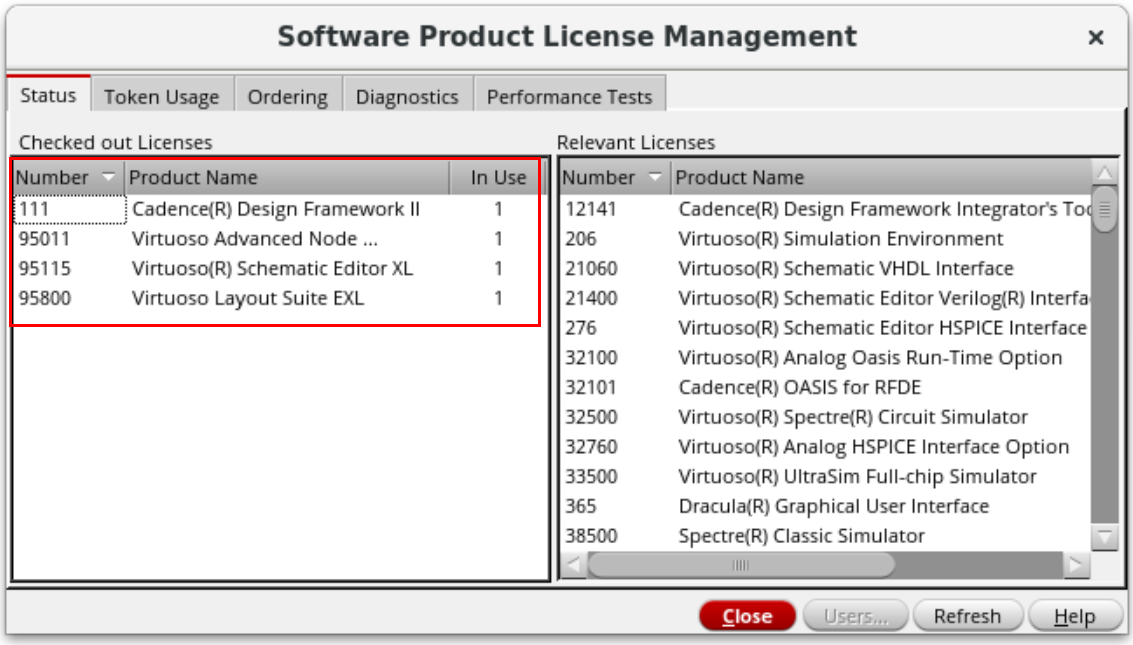
© 2023
All Rights Reserved.

Checking the License Status

You can check the license status using either of the following ways:

- Use the *Checked Out Licenses* pane of the *Status* tab.

  This pane lists the licenses that are currently checked out, and their count.



- Run the `lbuiGetCheckedOutLicenses` SKILL function to return a list of checked out licenses, which appear in the *Checked Out Licenses* pane.

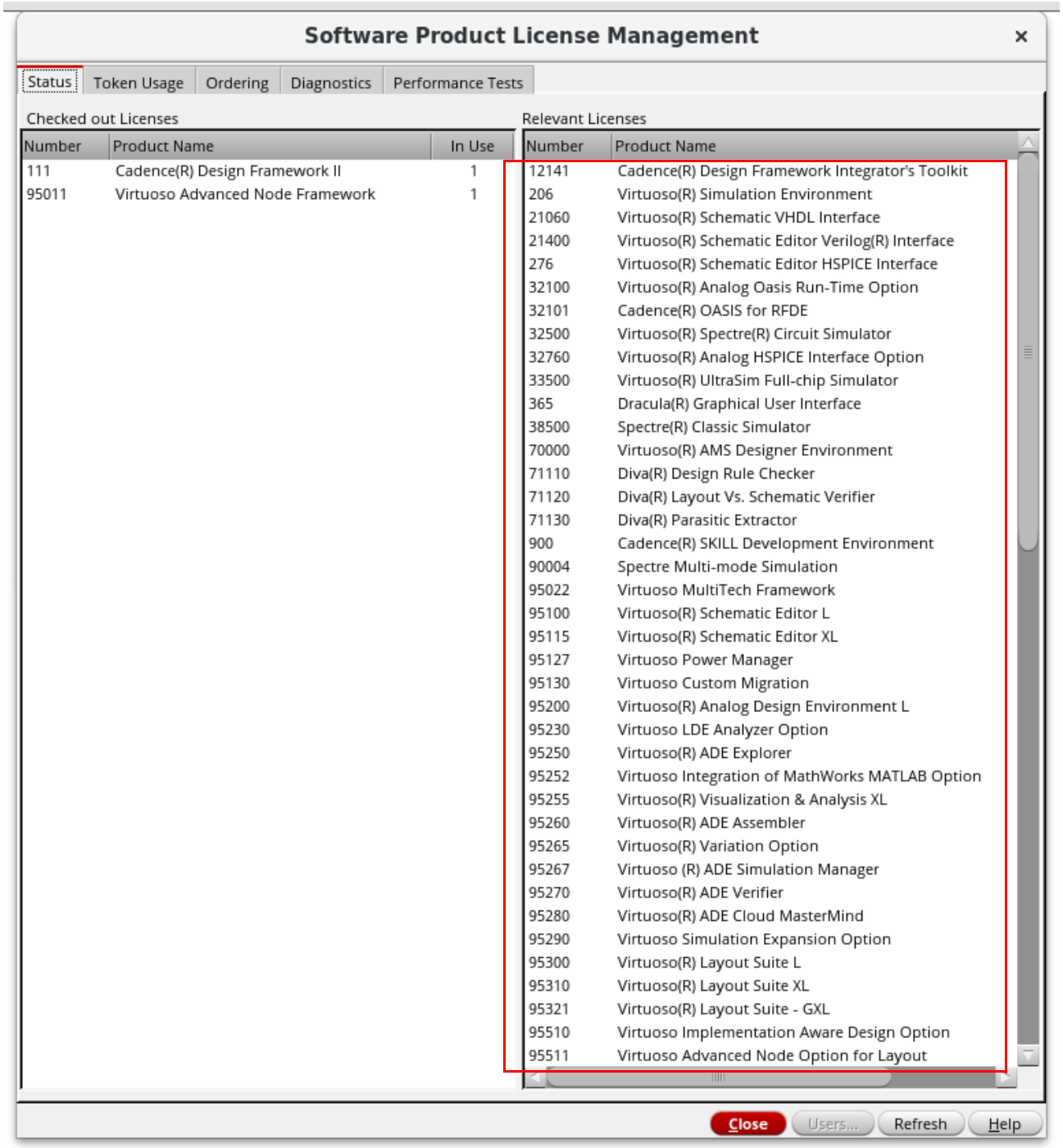
  The list contains each license feature name and the number of licenses (tokens) checked out. In addition, the license activity information gets recorded in the `CDS.log` file. This log file can be used later for creating reports to analyze the license usage.

  For more information on `lbuiGetCheckedOutLicenses`, refer to the *Virtuoso Studio Design Environment SKILL Reference*.

© 2023 All Rights Reserved.

Viewing Relevant Licenses

The *Relevant Licenses* pane on the *Status* tab lists the DFII licenses relevant to all of the tools incorporated into the workbench.



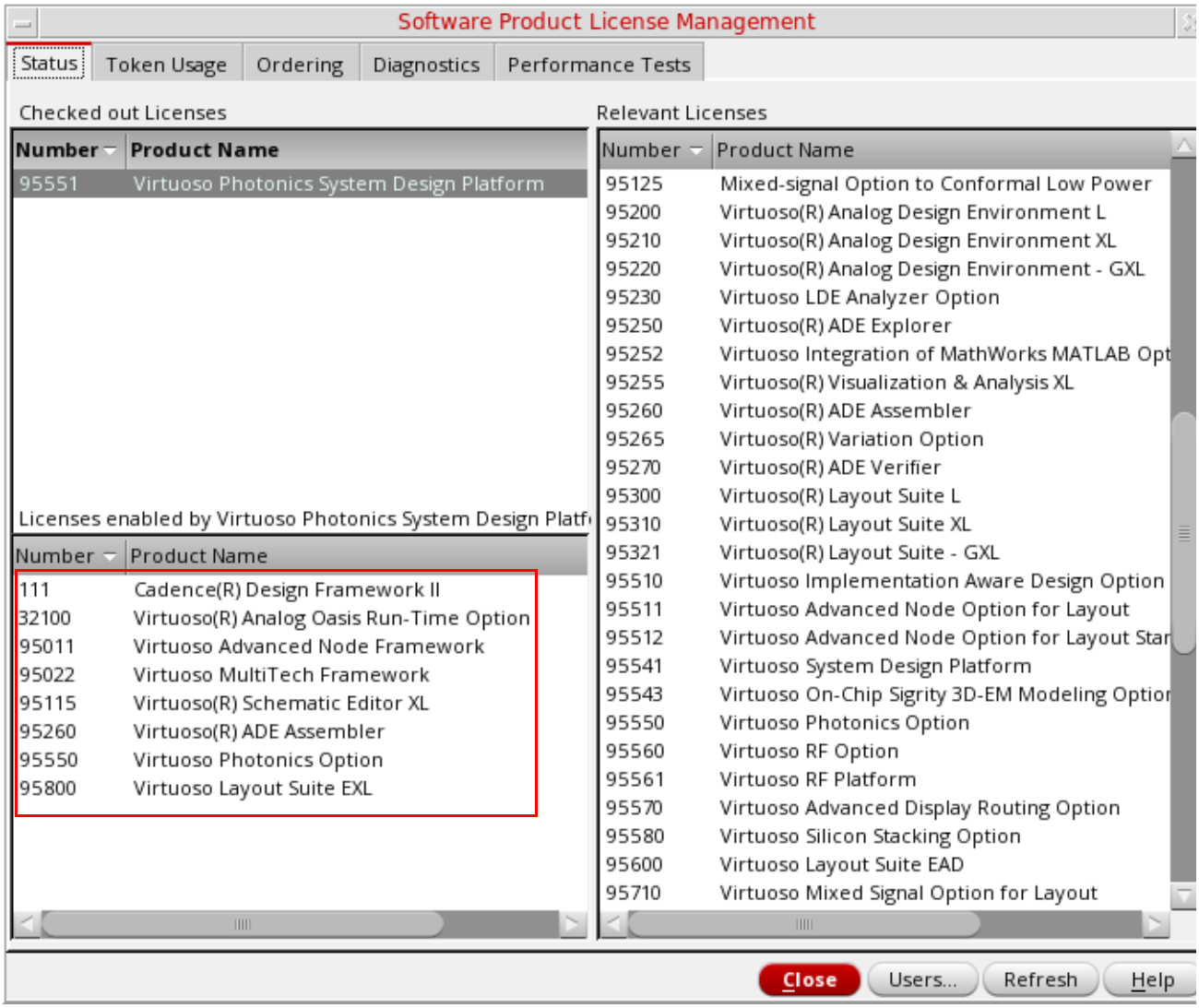
© 2023
All Rights Reserved.

Note: You can sort the licenses displayed in the *Checked out Licenses* and *Relevant Licenses* panes by clicking the arrow (▾) next to the *Number*, *Product Name*, or *In Use* column headers.

Viewing Solution-based Licenses

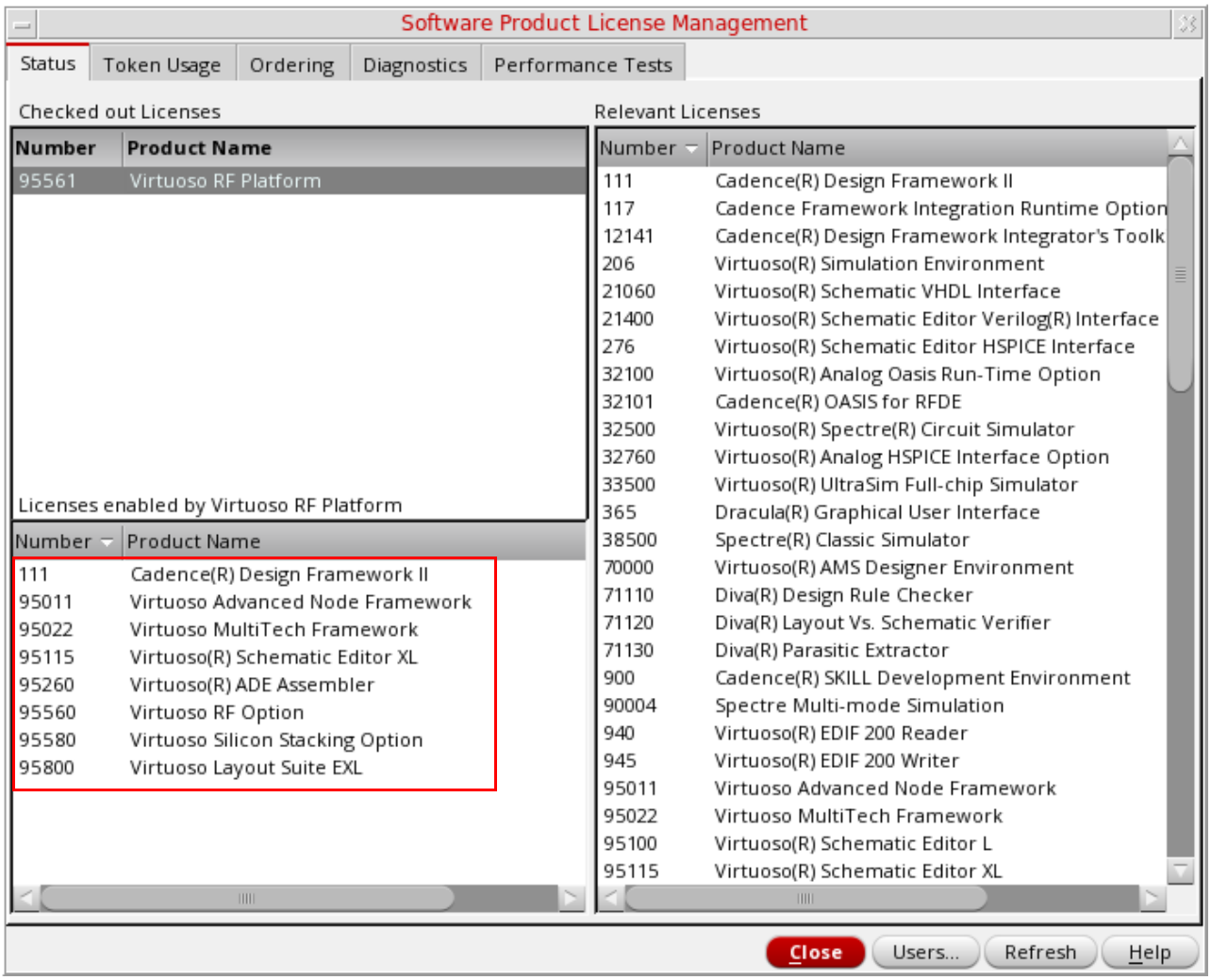
Licenses checked-out for a solution can be viewed when you set the specific environment.

The following licenses are checked out when the Virtuoso Photonics solution is enabled:



© 2023
All Rights Reserved.

The following licenses are checked out when the Virtuoso RF solution is enabled:



Adding Custom Products to the Status Tab

To add custom products and their license information to the *Status* tab, set the `CDS_PRODUCTS_DFII_LIC_UI` shell variable. The `CDS_PRODUCTS_DFII_LIC_UI` variable points to the file where the information about the custom products and license is stored in the following format:

```
<Product Number><separator><Product Name><separator><License Feature Name><endl>
where,
<separator> is ':'
and
is '\n'
```


© 2023
All Rights Reserved.

Here, the definition for the custom product and license information use this format in one line and any blank lines are ignored.

Refreshing the License Status

To update the *Checked Out Licenses* and *Relevant Licenses* panes, click *Refresh*.

This action will update the *Checked Out Licenses* and *Relevant Licenses* lists to show the current list of products being run, and the respective license count, along with any changes to the available license list since the initial access of the Software Product License Management form.

Displaying the License Usage Information

You can display the license status using either of the following ways:

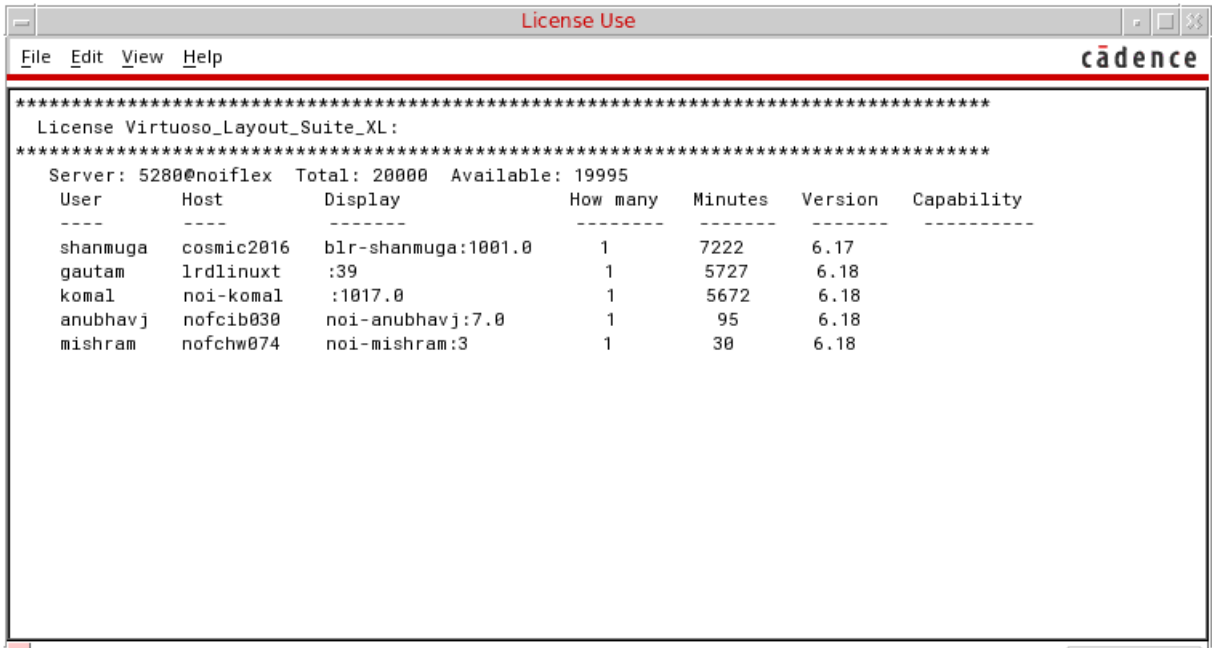
- Select a license from one of the lists on the Software Product License Management Form and click *Users* to display the License Use form.

  Note: The *Users* button is disabled by default. It gets enabled when you select a license in the Software Product License Management form and click *Refresh*.

  The License Use form displays the license usage information (for example: user logins, host names, display time, number of licenses, and license version) for the selected licenses, independent of the license-checkout status. Information in this form is read from the license servers that are defined in the `CDS_LIC_FILE` and is extracted from more

© 2023 All Rights Reserved.

than one license server. If you do not have any licenses for a selected feature in any license file, the License Use form will display total and available licenses as “0”.



Note: You can display the license usage information of more than one license in the License Use form. To do so, hold the CTRL key to select multiple licenses or hold the SHIFT key to select a contiguous range of licenses before clicking *Users*.

Hold the CTRL or SHIFT key and click to deselect any previously highlighted licenses before choosing a new license for viewing in the License Use form.

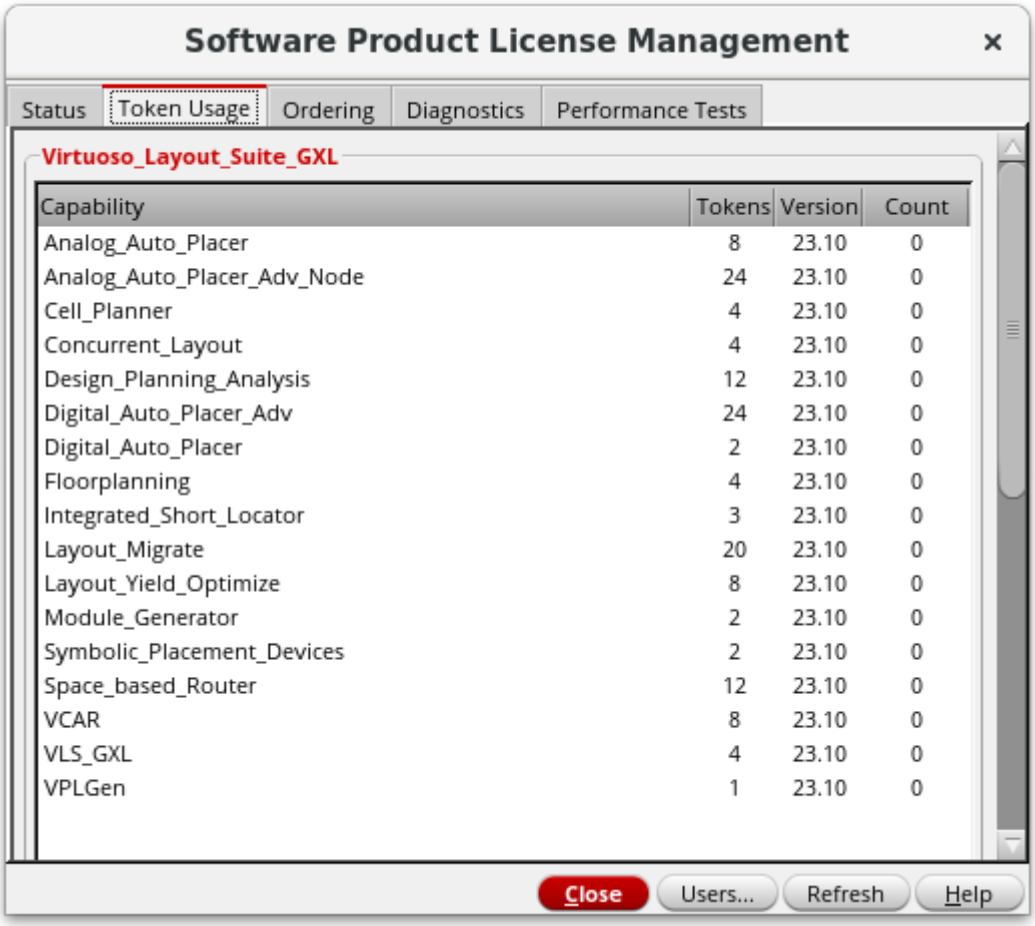
- Run the `lbuiPrintLicenseUseReport` SKILL function to display the license usage report (for example: user logins, host names, display time, number of licenses, and license version) in the CIW or the specified output log file.

  For more information on `lbuiPrintLicenseUseReport`, refer to the *Virtuoso Studio Design Environment SKILL Reference*.

© 2023 All Rights Reserved.

Software Product License Management Form: Token Usage Tab

The *Token Usage* tab displays the number of tokens required to run each capability of the Virtuoso Layout Suite GXL and the respective usage status.



Note:

- When an application that requires tokens is being launched, the *Total Tokens* and *Total available tokens* fields on the Token Usage form display the message “Collecting data from server”. This message is replaced with the token information once the launching process of the application completes.
- The L and XL product tiers have user, host, and display (UHD) based licenses. If you are using the same host and the same display, you can use the same L or XL license for multiple sessions of an application. However, the GXL tier uses job-based licenses, which are not shared. It means that each run of a GXL feature will check out the required number of tokens and multiple runs will require multiple sets of tokens. For more

© 2023 All Rights Reserved.

information on tokens, and licensing in general, see Advanced Features and Token Licenses

© 2023 All Rights Reserved.

Software Product License Management Form: Ordering Tab

The *Ordering* tab provides an interface for defining the license checkout order of the Virtuoso application tiers.

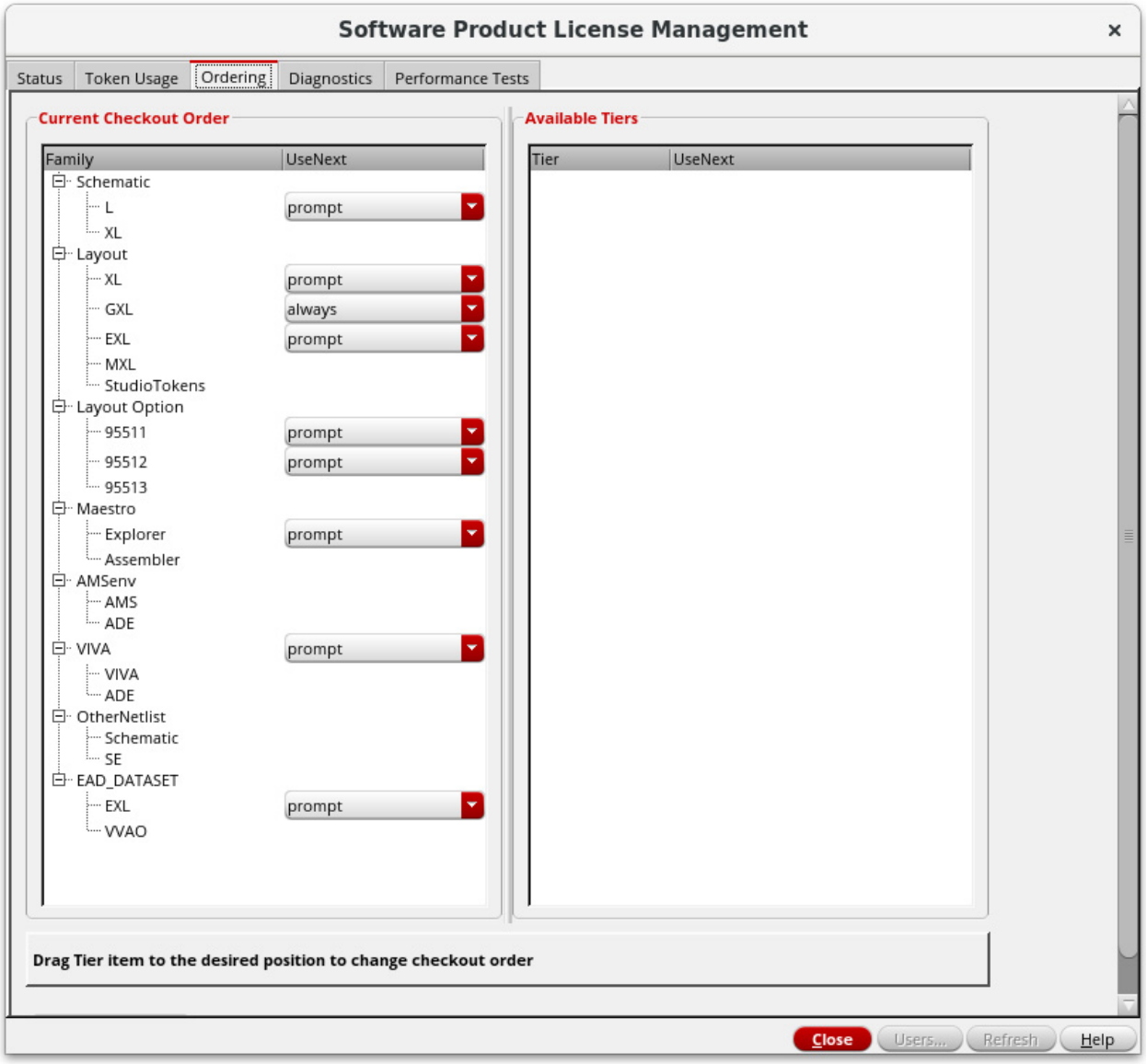
Each of the product families (Schematic, Layout, Layout Option, ADE, and VIVA) contains applications in a tiered packaging structure. Each higher-tier application provides additional features with more automated design assistance.

Only one license per product family tier is checked out at a time. Lower-tier applications require either their own tier license or a higher-tier license to run. For example, you can run VLS XL and VLS EXL in the same session with only a VLS EXL license checked out. If an attempt to check out the license of a lower-tier application fails, the application looks for a higher-tier license. You can set the license checkout order using the *Ordering* tab, as shown below.

© 2023 All Rights Reserved.

To view and use the displayed controls and buttons:

1. Click the *Ordering* tab in the Software Product License Management form. The *Ordering* tab has two panes: *Current Checkout Order* and *Available Tiers*.



- *Current Checkout Order:*

  Displays the list of product families along with their current checkout order and *UseNext* settings.

  Family: Lists the names of the product license families. For example, Schematic, Layout, Layout Option, ADE, AMSenv, VIVA, and EAD_DATASET.

© 2023
All Rights Reserved.

To change the checkout order of a product family, click and drag the required tier up or down to move it to the desired location.

Note: The GXL and EXL tiers in the Layout family cannot be separated; they must appear together in the checkout order.

Note: EAD licenses will automatically forward-enable Layout EXL.

UseNext: Displays the *UseNext* setting, which controls whether to check out the next available license for Virtuoso tier-based applications when the license required for a requested application is not available.

If a tiered application has an associated *UseNext* setting, a drop-down list appears next to its name. If you reorder a tier within the product family, the *UseNext* drop-down moves along with it.

The *UseNext* drop-down list provides the following options:

- *prompt:* Confirm before checking out a higher-tier license.
- *always*: Always check out the higher-tier license (as per selected checkout order).
- *never*: Never check out the higher-tier license and display an error message instead.

When *prompt* is set as the *UseNext* drop-down value and the license for a requested application is not available, the Next License dialog box displays, which lets you check out a higher tier license, subject to availability. Four options are offered: *Session*, *Skip*, *Always*, or *Never*.

- *Session*: Checks out a higher-tier license, if available, for the current session. If a higher-tier license is unavailable, the Next License dialog box will only be displayed once. However, a new virtuoso session will cause the dialog box to display again.

  The *Session* option does not override the setting in the local .cdsenv file and updates only the internal or in-memory value of the corresponding `UseNextLicense` .cdsenv variable. Because of this, the *Session* response is applicable for the current session only. However, if you want to restore the Next License dialog box settings in the same session, reset the `UseNextLicense` .cdsenv variable to `prompt` using the `envSetVal` function in the CIW or select *prompt* from the *UseNext* drop-down list.

- *Skip*: Skips the next license. For example, if L license is not available, and you click *Skip* in the Next License dialog box, the XL license will be skipped and an attempt to check out the GXL license will be made.

© 2023 All Rights Reserved.

- *Always:* Virtuoso will always try to check out the next license, if the requested license is not available. The Next License dialog box is not displayed again when virtuoso is restarted.
- *Never*: Virtuoso will never try to check out the next license, if the requested license is not available. The Next License dialog box is not displayed again when virtuoso is restarted.

The *Always* and *Never* options are stored in the following environment variables in ~/.cdsenv as soon as they are set.

For example,

```
license VLSXL_UseNextLicense string "prompt"/"always"/"never"
license VSEL_UseNextLicense string "prompt"/"always"/"never"
```

If a license is not available warning messages will be displayed in the CIW.

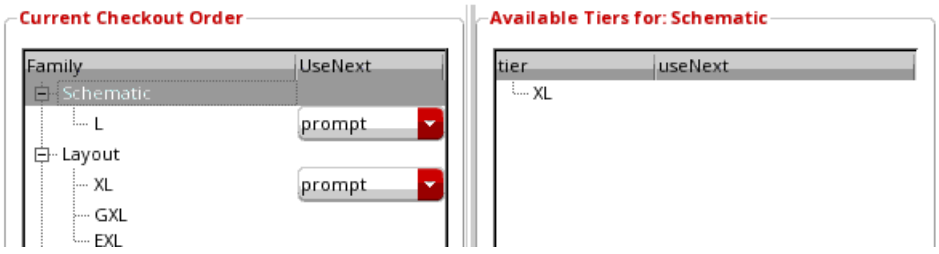
Note: The settings selected in the *Current Checkout Order* pane are reflected in the `.cdsenv` variables *`CheckoutOrder`* and *`UseNextLicense`*.

- *Available Tiers*:

By default, the *Current Checkout Order* pane displays all applicable tiers for each product family. However, you can choose to deselect a tier and remove it from the current checkout order list by dragging and moving it to *Available Tiers*.

Note: If there is only one tier in the family tree, you are not allowed to remove it from the *Current Checkout Order* list by dragging and moving it to *Available Tiers* list, because a family tree cannot be empty.

When you select a product family in the *Current Checkout Order* pane, you can see the current checkout order of its tiers in the left pane and the list of available (applicable) tiers in the right pane. For example, in the image below, the XL tier for the product family Schematic has been moved to the *Available Tiers for: Schematic* pane.



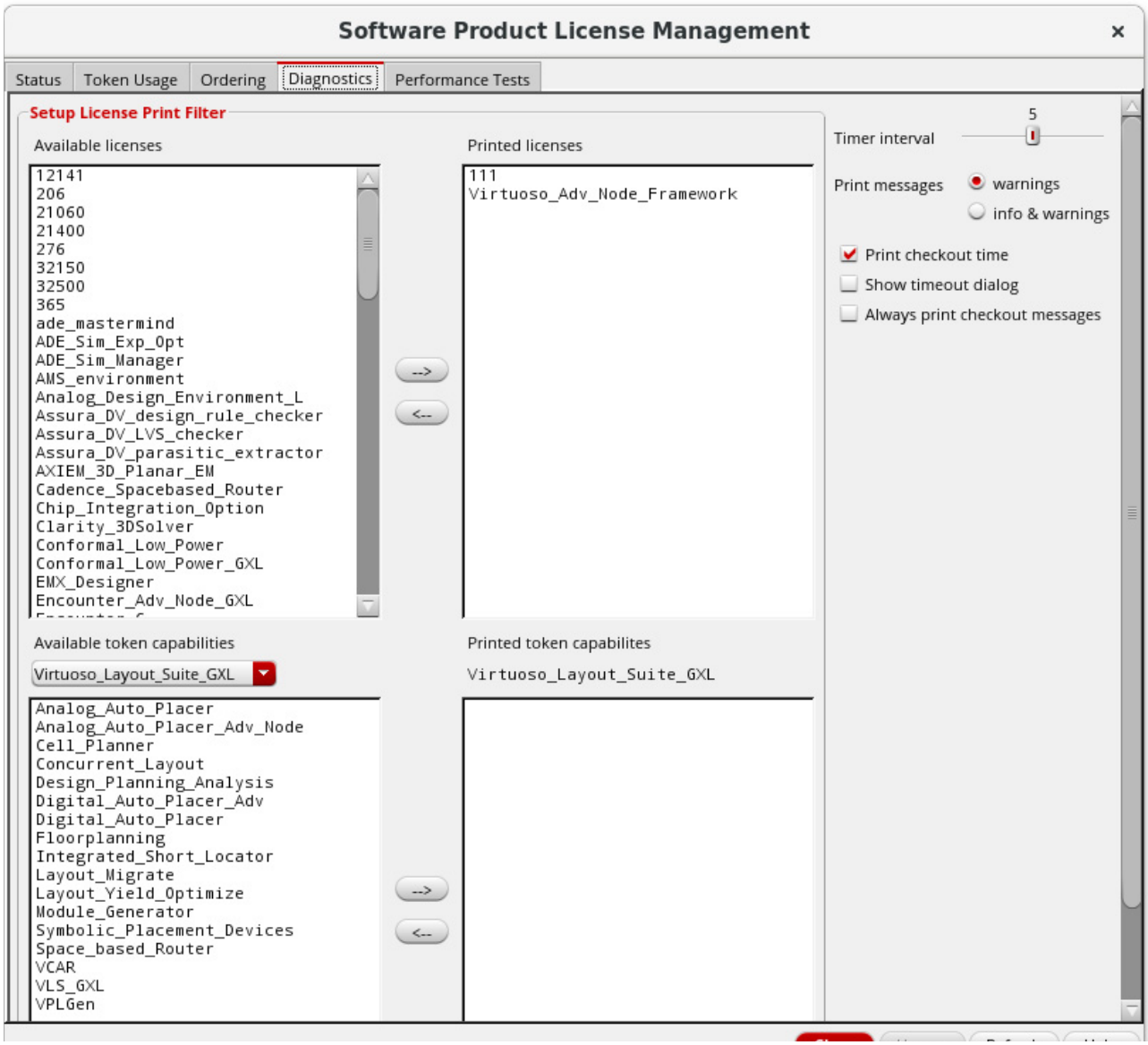
2. Click *Save to File* to save the selected options in the `.cdsenv` file. You can also edit your preferences in the `.cdsenv` file.

© 2023
All Rights Reserved.

Software Product License Management Form: Diagnostics Tab

The *Diagnostics* tab provides an interface for setting the parameters needed for license-checkout reporting. For example, you can set these parameters to report whether a license check is taking too long or if a particular license or token has been checked out. By using the options available in this tab, you can set the values of the license diagnostic variables without needing to remember the shell variable names.

➡ Click the *Diagnostics* tab in the Software Product License Management form to view and use the displayed panes and controls.



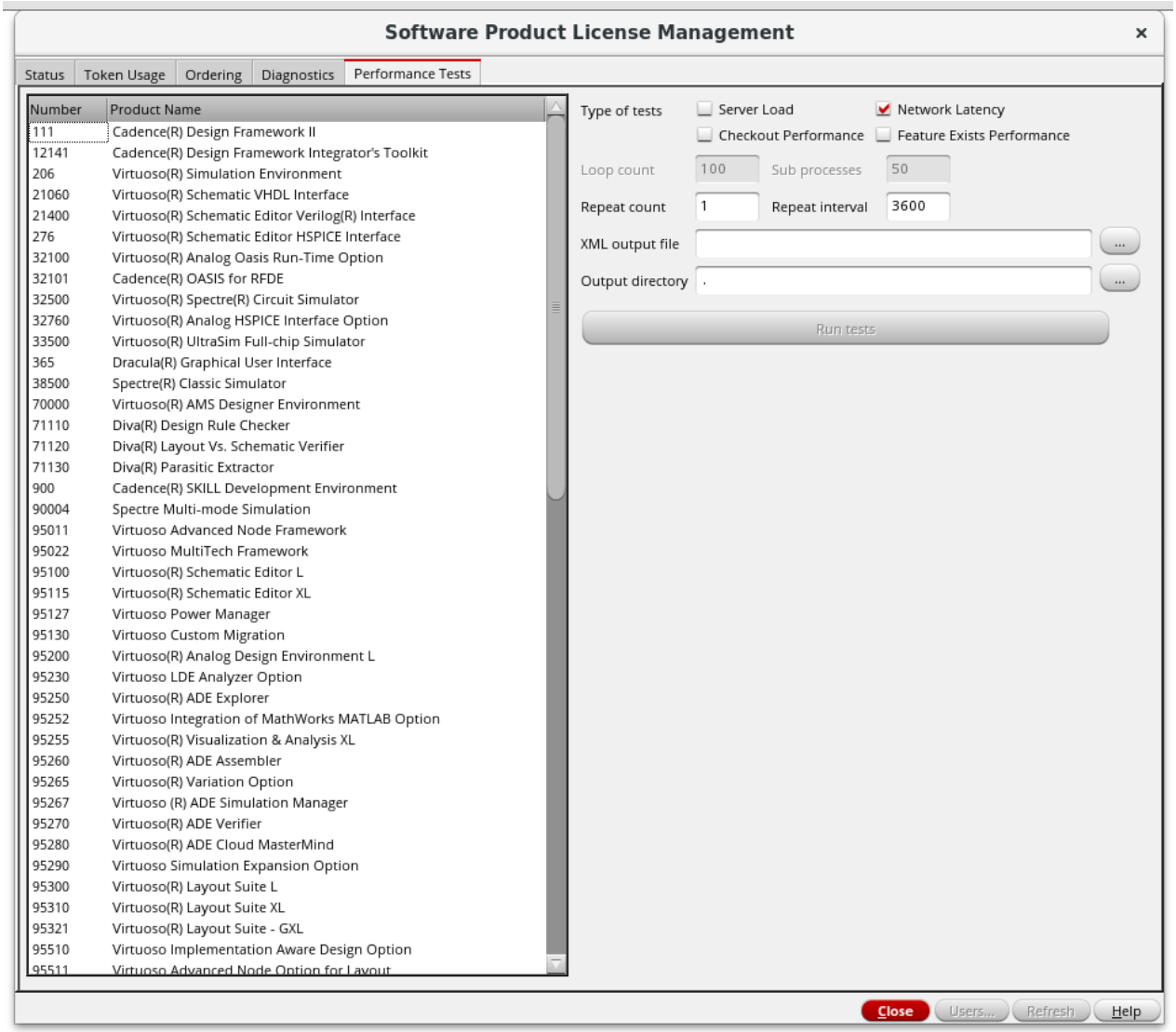
© 2023
All Rights Reserved.

Software Product License Management Form: Performance Tests Tab

In an environment where you have licenses served across multiple servers spread over a network, there can be significant delays during the application start up because Cadence applications search all the license servers to acquire a license.

In such situations, you can run a diagnostic test from the *Performance Tests* tab to test the license checkout performance on a feature-by-feature basis and across each server. This tab is an interface to the `perf_test` utility.

➡ Click the *Performance Tests* tab in the Software Product License Management form to use the displayed controls and buttons.



© 2023
All Rights Reserved.

For testing the network latency, selecting the Cadence Design Framework II license is sufficient. However, if you suspect that a particular license feature is taking more time to check out, you can select the required license feature name for testing.

© 2023 All Rights Reserved.

Product Tier Features

The following table shows the feature details included in each application.

Virtuoso Layout Features

Virtuoso Layout Features

In the table:

- ✓ Specifies that the feature is included with the Base Product.
- ✓+ Specifies that the feature requires both Base Product and Option.

Virtuoso Layout Features	New in IC23.1	VLS XL 95310	VLS EXL 95800	VLS MXL 95810	VLX GXL Option 95323
Schematic Driven Layout		✓	✓	✓	
Constraints and Design Intent		✓	✓	✓	
Schematic Assistant	New	✓	✓	✓	
Incremental Check Against Source	New	✓	✓	✓	
Multi-threaded Connectivity Extractor	New	✓	✓	✓	
Wire and Bus Editing	New	✓	✓	✓	
Auto Via	New	✓	✓	✓	
GroupArray	New	✓	✓	✓	
In-design DRC with iPegasus	New	✓	✓	✓	
In-design Metal Fill with iPegasus	New	✓	✓	✓	
Design Review Editor	New	✓	✓	✓	
Modgen		✓+	✓	✓	+
Symbolic Placement of Devices		✓+	✓+	✓	+
Concurrent Layout Editing		✓+	✓	✓	+

© 2023 All Rights Reserved.

Virtuoso Layout Features	New in IC23.1	VLS XL 95310	VLS EXL 95800	VLS MXL 95810	VLX GXL Option 95323
Simulation Driven Routing			✓	✓	
In-design EM/IR + RC			✓	✓	
Partial Layout Resimulation	New		✓	✓	
Capture and Replay	New		✓	✓	
Row based layout flow			✓	✓	
Auto Placer and Fill			✓	✓	
Assisted Routing		✓+	✓+	✓	+
Copy Route	New		✓	✓	
Design Planner			✓+	✓	+
Congestion Analysis			✓+	✓	+
Electromagnetics Assistant			✓	✓	
Device Level Auto Routing	New			✓	
GigaPlace and NanoRoute Integration	New			✓	
Spine style routing for memory	New			✓	
Chip + Block Assembly Auto Routing	New			✓	
Unified Routing Assistant	New			✓	
Multi-fabric and Multi-Tech support	New			✓	
EMX and Clarity Integration for RF	New			✓	
Layout Reuse and Migration	New			✓	
Analog_Auto_Placer		✓ +	✓ +	✓	+
Digital_Auto_Placer		✓ +	✓ +	✓	+
Digital_Auto_Placer_Adv			✓ +	✓	+

© 2023 All Rights Reserved.

Virtuoso Layout Features	New in IC23.1	VLS XL 95310	VLS EXL 95800	VLS MXL 95810	VLX GXL Option 95323
Floorplanning		✓ +	✓ +	✓	+
Integrated_Short_Locator		✓ +	✓ +	✓	+
Slotting		✓ +	✓	✓	+
Space_based_Router		✓ +	✓ +	✓	+
VPLGen		✓ +	✓	✓	+

ADE Product Family Features

The following table lists the features available in the ADE product family:

In the table:

- ✓* Specifies not all features within the GUI are available to OASIS integrated (third-party) simulators. Contact Cadence CPG Marketing team for individual situations.
- ✓* Specifies that the feature may also require Legato Reliability Solution license.

Virtuoso ADE Suite Features	New in IC23.1	ADE Explorer 95250	ADE Assembler 95260	ADE Verifier 95270	Virtuoso Variation Option 95265
Single Test-bench support		✓	✓		
Device Instance Parameters and Design Variables Sweep Support (Parameterization)		✓	✓		
Built-in variation analysis (Corner, Monte-Carlo, Sensitivity, Reliability analysis)		✓	✓		

© 2023 All Rights Reserved.

Virtuoso ADE Suite Features	New in IC23.1	ADE Explorer	ADE Assembler	ADE Verifier	Virtuoso Variation Option
		95250	95260	95270	95265
Filtering in outputs setup, corners setup and results, filtering in dataview or setup assistant		✓	✓		
Spec-driven measurements with pass or fail results display		✓	✓		
Waveform plotting with extensive capabilities in ViVA incl plotting templates support		✓	✓		
On schematic waveform thumbnails/info balloons		✓	✓		
Integration with Cadence Spectre Simulation Platform		✓	✓		
Third-party simulators integration support		✓*	✓*		
Measurements across Sweeps, Corners and Dimensions		✓	✓		
Assertions and Device Checks Support (SPECTRE ONLY)		✓	✓		
HTML Spec Sheets for design review		✓	✓		
Scripting support with OCEAN or Mastero APIs (simpler vs OCEAN)		✓	✓		

© 2023 All Rights Reserved.

Virtuoso ADE Suite Features	**New in IC23.1**	**ADE Explorer**	**ADE Assembler**	**ADE Verifier**	**Virtuoso Variation Option**
		95250	95260	95270	95265
Mathworks Matlab expression support,data post-processing/script development in Matlab environment		✓ +95252	✓ +95252		
Faster MC post-processing (histograms, scatter plots, etc.) with new db for statistical parameter data storage	New	✓	✓		
Distributed processing with farms (LSF, SGE, etc.) support		✓	✓		
Parasitic Resimulation w/ Extracted View (including Smart View)		✓	✓		
Real-time circuit tuning with dynamic waveform display		✓			
Multiple test-bench support			✓		
Data View/History Assistant (including Merge History support)			✓		
Reconnect simulations/results when Virtuoso/ADE exits	New		✓ +95267		
Distributed results plotting and re-evaluation	New		✓		
Support for pre-calibration scripting			✓		
Run Plans for mini-verification runs			✓		

© 2023 All Rights Reserved.

Virtuoso ADE Suite Features	New in IC23.1	ADE Explorer	ADE Assembler	ADE Verifier	Virtuoso Variation Option
		95250	95260	95270	95265
Waveform specs to autocheck wave repsonses			✓		
Global and Local Circuit Optimization (Classic)			✓		
Partial Layout device extraction resimulation w/ EAD flow			✓ +95800		
Spectre Interactive Environment (SIE)			✓		
Worst Case Corners Analysis			✓		
Advanced Statistical/Optimization/Sensitivity Support					
High-sigma MC Analysis to identify the worst-samples using Spectre FMC (3-6s+)	New	✓ +	✓ +		+
Sensitivity Accuracy and Mismatch Contribution analysis incl Mismatch tuning		✓ +	✓ +		+
Assisted Yield Improvement Flow			✓ +		+
New Advanced AI Optimization (Use Algorithms/Integrate)	New		✓ +		+
Legato Reliability Solution					
Electro-thermal analysis		✓**	✓**		
Advanced aging analysis		✓**	✓**		
Analog Defect simulation			✓**		
ADE Verifier Electrical Verification					

© 2023 All Rights Reserved.

Virtuoso ADE Suite Features	New in IC23.1	ADE Explorer	ADE Assembler	ADE Verifier	Virtuoso Variation Option
		95250	95260	95270	95265
Creation of analog verification plans with requirements/specs				✓	
Running large regression setups from Verifier				✓	
Setup library assistance to define sweep, corner, simulation setting setups and share across multiple Assembler/Verifier views				✓	
Analog verification coverage report with spec tracing across users				✓	
Connection to Cadence's Verisium Manager for mixed-signal coverage				✓	
Results traceability to support ISO requirements				✓	
Support for composite requirements to for complex verification setups and results slicing according to verification spaces	New			✓	
Verification snapshots				✓	
ADE Run Modes <Sweet Spots>					
Classic ICRP < 100 parallel jobs		✓	✓	✓	
LSCS <500 parallel jobs			✓	✓	
ADE Simulation Manager < 5000 parallel jobs	New		✓ +95267	✓ +95267	

© 2023 All Rights Reserved.

Virtuoso ADE Suite Features	New in IC23.1	ADE Explorer	ADE Assembler	ADE Verifier	Virtuoso Variation Option
		95250	95260	95270	95265
Expand to more than 400 jobs running in parallel	New		✓ +95290	✓ +95290	

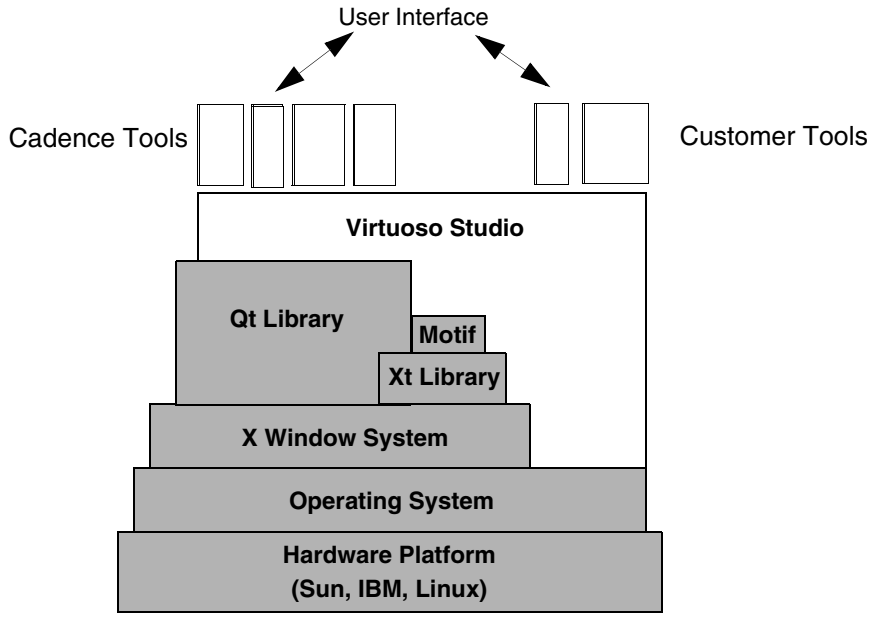
© 2023 All Rights Reserved.

Hardware and Software Requirements

For hardware and software requirements needed to run the Virtuoso Studio, see *Hardware and Software Requirements* in the *Virtuoso Studio Design Environment User Guide*.

Software Layers

The figure below shows how the Virtuoso Studio resides on top of several layers of software (bottom layer is your hardware platform).

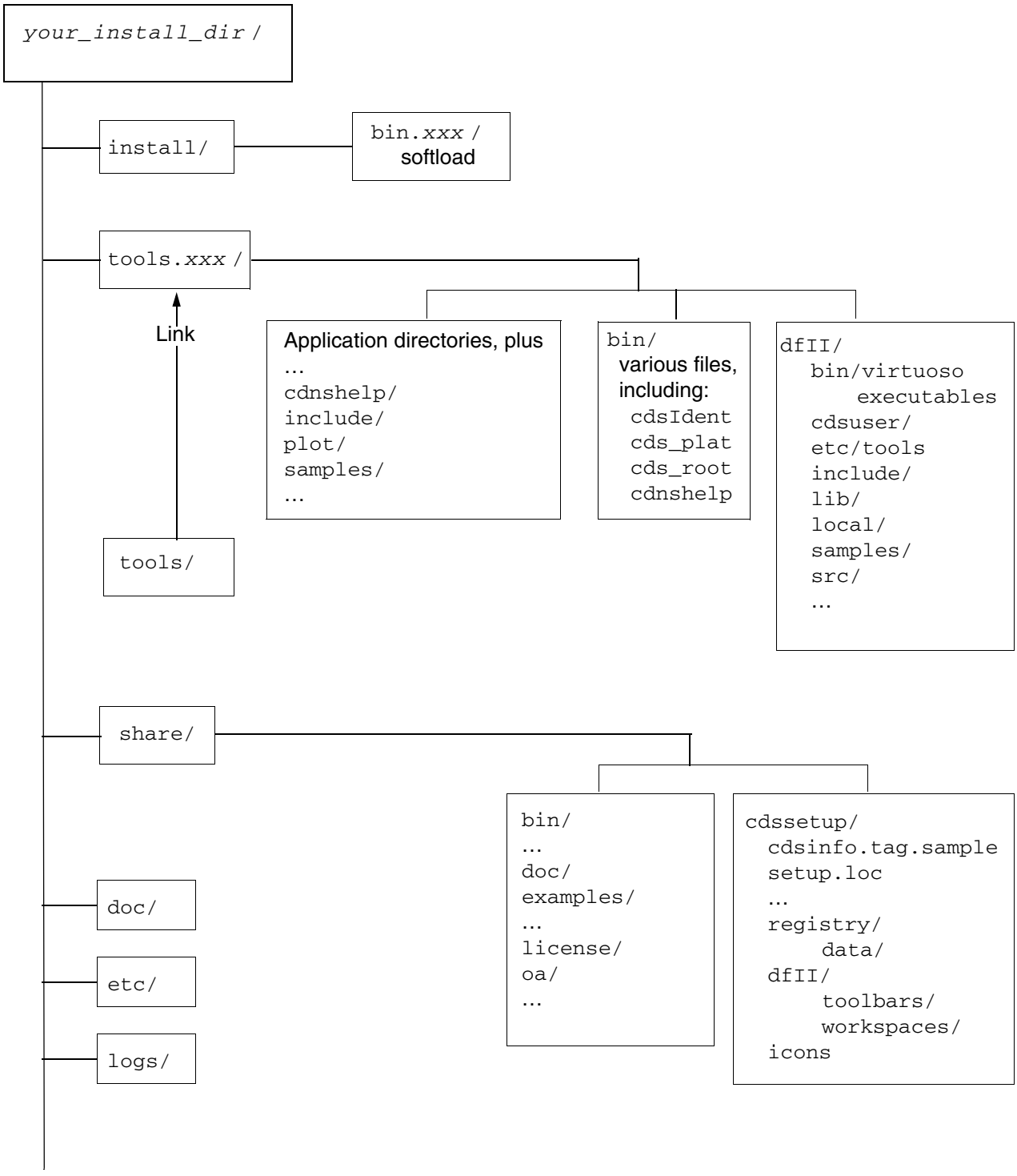


For More Information	Refer to
Installation	*Cadence Installation Guide*
Licensing	*Cadence License Manager*
File locking	*Cadence Application Infrastructure User Guide*
Plotting	*Plotter Configuration User Guide*
Design flow	Your Cadence applications engineer (AE)

© 2023 All Rights Reserved.

Virtuoso Studio Design Environment Hierarchy

The software uses the following directory hierarchy to store executables and data files.



© 2023 All Rights Reserved.

The *your_install_dir*/tools/dfII directory includes the directories listed below. The files in these directories depend on which Cadence® products are installed.

bin	Virtuoso Studio design environment software executables for your platform.
bin.*xxx*	Link to bin directory.
cdsuser	Sample home directory environment setup files such as .Xdefaults, .cdsinit, .cshrc, .mwmrc, .simrc, .xinitrc, and arrowKeys.map.
etc/tools	Directories corresponding to particular applications or core pieces of infrastructure such as layout, schematic, and cdba. Each directory contains a default .cdsenv file and might also contain other sample files.
include	C header files to go with licensed Cadence products.
local	CAD group customization files and site-specific configuration files. **Note:** Installing new software does not overwrite this directory.
lib	Programming libraries.
samples	Sample user files. Some of the samples include the following: .cdsenv — Sample environment variable file local — Sample customization files plot — Schematic plotting files techfile — Technology files and libraries tutorials — Tutorials for various products
script	Configuration scripts.
src	More customization files.

© 2023 All Rights Reserved.

Virtuoso Studio Design Environment Executables

Virtuoso Studio design environment executables include:

- **`virtuoso`**: All Virtuoso software executables are integrated into one executable called `virtuoso`. Both 32- and 64-bit platform variants exist for the `virtuoso` workbench.
- **`dbAccess`**: `dbAccess`is a non-graphical UNIX command-line executable for using the core SKILL functionality and accessing DFII OpenAccess design and technology file data. You can also use this executable to debug Pcells and verify that they run in other environments and have no dependencies on product packages or licenses. `dbAccess` supports `db`, `dd`, `rod`, and `tech` functions. Since it does not have a graphical user interface, it does not support `hi` and `ge` functions.

  Command Switches

  `dbAccess [{-32 | -64 | -32only | -64only | -3264 | -6432}] [-quiet3264][-debug3264] [-plat` *`platform`*`] [-v3264] [-help3264] <[-load` fileName] [-cdslib *`fileName`*`]>`

  Description

`-32`	Select the 32-bit version; if the 32-bit version is not available, print a warning message and try to launch the 64-bit version.
`-64`	Select the 64-bit version; if the 64-bit version is not available, print a warning message and try to launch the 32-bit version.
`-32only`	Select the 32-bit version; if the 32-bit version is not available, print an error message and exit.
`-64only`	Select the 64-bit version; if the 64-bit version is not available, print an error message and exit.
`-3264`	Select the 32-bit version; if the 32-bit version is not available, print an info message and try to launch the 64-bit version.
`-6432`	Select the 64-bit version; if the 64-bit version is not available, print an info message and try to launch the 32-bit version.
`-quiet3264`	Suppress the warning/error/info messages of the -32/-32only/-3264 and -64/-64only/-6432 options. -debug3264 Print the environment updated by the wrapper and the command launched.
`-plat` *`platform`*	Override the default platform selection when the tool is launched from the `<install_root>/bin` directory.

© 2023 All Rights Reserved.

`-v3264`	Print the wrapper's version string
`-help3264`	Display detailed help of `dbAccess` generic options.
`-load`	Loads and executes the specified SKILL file.
`-cdslib` *`fileName`*	Reads the specfied filename for library definitions. By default `dbAccess` reads the cds.lib file in the current working directory.

If you do not provide a command-line option, `dbAccess` switches to interactive mode and accepts commands from the keyboard or `stdin`.

In addition, `dbAccess` accepts the Virtuoso command-line options, such as:

`-V`	Displays Cadence release version.
`-W`	Displays Cadence release subversion.

However, it ignores options that require a graphical user interface.

Example

Suppose you have a SKILL file, `listLibs.il`, which includes commands to display the libraries specified in the `cds.lib` file. If a `cds.lib` file is not present, the default `cds.lib` from the installation directory is used.

The file contents are as follows:

```
foreach(lib ddGetLibList() printf("%s\n" lib~>name))
```

Now, if you run the `dbAccess` command and loads this SKILL file, the following output is displayed on a terminal window:

```
dbAccess -load listLibs.il
Virtuoso Framework License (111) was checked out successfully. Total checkout
time was 0.64s.
cdsDefTechLib
basic
US_8ths
rfLib
rfExamples
ahdlLib
```


© 2023 All Rights Reserved.

```
analogLib
functional
```

The above output is generated using the default `cds.lib` from the installation directory.

© 2023 All Rights Reserved.

Virtuoso Studio Design Environment Licensing Setup

The key licensing characteristics of the Virtuoso Studio design environment are outlined in the following table.

Licensing Characteristic	Virtuoso Studio Design Environment
Licensing software	SoftShare
Licensing DAEMON	`cdslmd`
License file	`license.`*`HOSTID`*
Path to DFII	*`your_install_dir`*`/tools/dfII`
Path to the license directory	*`your_install_dir`*`/share/license`
Variable for license file	See the *Cadence License Manager*
File locking	CLS

Note: Licenses from several license servers can be used if there are multiple independent license servers configured.

Important

All Cadence software requires Cadence licenses. To specify your license file, set the `CDS_LIC_FILE`. If a license cannot be found on any of the servers specified in the `CDS_LIC_FILE`, the license search will continue to look through the `LM_LICENSE_FILE`. If you do not want a license search to continue beyond the `CDS_LIC_FILE`, you need to set the `CDS_LIC_ONLY` environment variable which will instruct Virtuoso to ignore the `LM_LICENSE_FILE`.

For more information about license files and variables, see the *Cadence Installation Guide* and *Cadence License Manager*.

© 2023 All Rights Reserved.

Frequently Asked Questions Related to Virtuoso Studio Licensing

The following table answers frequently asked licensing questions:

Question	Answer
In interactive mode, when are licenses checked in?	Licenses are checked in automatically when you exit an application.
Do all Cadence products support global `TIMEOUT`?	Yes. `TIMEOUT` sets the time after which an inactive license is reclaimed by the vendor daemon. You can amend the default `TIMEOUT` setting in the options file to specify how long it is before a *product* license times out. The value entered will be valid as long as it exceeds the minimum setting (3600 secs). `TIMEOUTALL`, which applies a timeout to all features is also supported in DFII products. For example, `TIMEOUT` *featurename seconds* `TIMEOUTALL` *seconds*
What happens to my work and any open windows when products lose their licenses?	When you execute the next command, it will re-check out the license automatically. You will only notice an issue if the license you need (the license that timed out) is no longer available.
After startup, how often does a product check for a valid license?	The license-checking interval depends on the product. In general, interactive products (for editing) check at the start of every editing command and batch-mode products check every five minutes.
Can I wait for a license?	No. You cannot wait for a license.

© 2023 All Rights Reserved.

Question	Answer
If prolonged batch jobs require multiple licenses, are all licenses reserved or checked out up front?	This depends on the application.

© 2023 All Rights Reserved.

© 2023 All Rights Reserved.

2

Setting Up Virtuoso Studio

Each time the application software runs, the application loads variables in the following order:

1. Resources in the resource database.

   The resources are taken directly from the `.Xdefaults`.

2. Variables in the <*app*>`_defaults` file, where *app* is the application or group name.
3. Variables set by the application
4. Variables set by the <*your_install_dir*>`/tools.`<*plat*>`/`<*app_dir*>`/local/.cdsenv` file
   - <*your_install_dir*> is the directory that contains the software
   - <*app_dir*> is the directory containing the Cadence® application files
5. Environment variables
6. Program loads all `.cdsenv` files in the following order if the `CDS_LOAD_ENV` environment variable is not set:
   - <*your_install_dir*>`/tools.`<*plat*>`/dfII/etc/tools/`*application*`/.cdsenv`
   - <*your_install_dir*>`/tools.`<*plat*>`/dfII/local/.cdsenv`
   - `$HOME/.cdsenv`
7. Program searches the following locations in order and loads variables from the first `.cdsinit` file encountered:
   - <*your_install_dir*>`/tools.`<*plat*>`/dfII/local`
   - current working directory: `.`
   - your home directory

Individual user settings override the variables set by the application and any group customization. You can copy the `.cdsenv` site file to your local directory and customize it or

© 2023 All Rights Reserved.

choose *Options – Save Defaults* in the Command Interpreter Window (CIW) to save variable settings to a `.cdsenv` file.

Related Topics

Saving New Default Values

Quick Start: Linux and Unix Environments

This section outlines the quick start procedure for Linux and Unix environments running X Windows graphics software.

You may need to use a `root` login to modify files or change permissions.

1. Set up a user account.
2. Modify and apply your search path as follows:

   `set path = ( <your_install_path>/bin/<app> $path )`
3. Start the software by typing:

   `executable &`

   where *`executable`* is the command you want to run (for example, `virtuoso`).

   The CIW appears after a short initialization period.
   You must set up plotters before you can plot a design.

Related Topics

Setting Up a User Account

Virtuoso Studio Design Environment Executables

Plotter Configuration User Guide

Quick Start Guide to Setting Up a Plotter

© 2023 All Rights Reserved.

Configuration and Startup Procedures

The tasks below provide a guide to the full configuration and startup procedure for your software. This document assumes you have copies of the default initialization files from `<your_install_dir>/tools.<plat>/dfII/cdsuser/.*` in a local directory and that you are logging in through `dtlogin` (perhaps from `/usr/dt/bin`).

Setting the Installation Path for Cadence Tools

Cadence tools do not require any environment variable to specify their installation directory.

Below we only provide example using some variables for tools that require environment variable to specify the installation directory. You can select any name you want for these variables in your environment.

Cadence tools are typically installed in their own directory hierarchy. You should use different UNIX environment variables to represent each release hierarchy. The following example shows how to set installation paths in a `.cshrc` file (you must modify these paths to match your installation).

For information regarding setting environment variables for specific tools, refer to the documentation for that particular tool.

```
#*****************************************************************#
# Set path to DFII install directory
set ICHOME = /cdsIntall/cds/IC231
#
```


© 2023 All Rights Reserved.

```
# Set path to ICC install directory
setenv ICCHOME /cdsInstall/cds/iccraft112
#

#
# Set path to Innovus install directory
setenv INVSHOME /dsmInstall/INVS191
#
#*********************************************************************#
#
# Set paths to Cadence executables
#(In this example, variable names are assigned for each tool path,
# then set in the set path statement)
#
set dfiiPath = ( $ICHOME/bin )
set iccPath = ( $ICCHOME/bin )
set invsPath = ( $INVSHOME/bin )

#
set path=( ~/bin \
/usr/bin /bin
$dfiiPath \
$iccPath \
$assuraPath \
$invsPath \
    )
#*********************************************************************#
# Set paths to license files
setenv CDS_LIC_FILE $CDSHOME/cdsbin/lmtools/license.dat/license.dat

setenv CDS_LIC_ONLY 1

#This makes license check faster both for Cadence and non-Cadence tools to segregate
Cadence server in the CDS_LIC_FILE and keep only the non-cadence servers in the
LM_LICENSE_FILE.
#*********************************************************************#
# END of the .cshrc file
#*********************************************************************#
```

Setting Up a User Account

To set up accounts for new users, you must already have created a login and home directory for each user. The <*your_install_dir*>/tools.<*plat*>/dfII/cdsuser directory contains sample home directory environment setup files.

Next, follow the steps outlined in Copying Sample Files to Your Home Directory.

Once you have set up a user account, you can modify your search path and run the software as described in Quick Start: Linux and Unix Environments.

Copying Sample Files to Your Home Directory

1. Copy the sample files to your home directory.

© 2023 All Rights Reserved.

You can use commands similar to the following:

```
cd
cp <your_install_dir>/tools.<plat>/dfII/samples/.??* .
cp <your_install_dir>/tools.<plat>/dfII/cdsuser/.??* .
```

The copy command (`cp`) will overwrite any existing files of the same name in your directory.

2. Go through each section of the sample files and customize them for your site. Be sure to replace `/cdsdir` with <*your_install_dir*>.

Setting the Focus Correctly

On KDE, Red Hat Enterprise Linux, and Gnome, a newly opened window might pop up under a window or on the desktop and is not immediately visible. You can correct this as follows.

In the KDE window manager:

1. Open KDE Control Center or type `/usr/bin/kcontrol` on LINUX command prompt to start the KDE Control Center.
2. Select *Desktop — Window Behavior*.
3. Select the *Advanced* tab.
4. Set *Focus stealing prevention level* to `None` or `Low`.

On LINUX:

1. Open the *System Settings* form by running the following command:

   ```
   LINUX> /usr/bin/systemsettings
   ```

2. Click the *General* tab and then *Window Behavior*.
3. Set *Focus stealing prevention level* to `None` or `Low`.

On Gnome:

1. Open the Gnome Configuration Editor window by entering the `gconf-editor` command at the terminal.
2. Navigate to *apps — metacity — general*.
3. Select `new_windows_always_on_top`.
4. Set the *`focus_new_window`* variable to `strict`.

© 2023 All Rights Reserved.

The default value of *Single-click to open files and folders* causes double-click actions to be invoked with a single click.

For information about setting the font options, see *Viewing the Font List* in the *Virtuoso Studio Design Environment User Guide*.

© 2023 All Rights Reserved.

Specifying Cadence Environment Variables

You can specify any number of the following environment variables.

Cadence Environment Variables

Variable	What You Can Specify
CDS_DEFAULT_BROWSER	Sets the initial value of the `ui .cdsenv` variable `webBrowser` to this value. If set, this is overridden by the value of `ui.webBrowser` in the `.cdsenv` file.
CDS_LOG_PATH	A colon-separated path of directories (ordered by preference) in which to put the log file. If none of these directories exist or they are all not writable, the default `$HOME` directory is used.
CDS_LOAD_ENV	A customized search order for `.cdsenv` using one of the following keywords: ■ `False` loads neither `~/.cdsenv` or `CWD/.cdsenv` ■ `CWD` loads only `/.cdsenv` from the current directory ■ `addCWD` loads `~/.cdsenv` and then loads `CWD/.cdsenv` ■ `CWDElseHome` loads `CWD/.cdsenv` if it exists, or `~/.cdsenv`, if it exists ■ `CSF` uses the Cadence Setup Search File mechanism to find the `.cdsenv` files to load For more information, see "Specifying a Search Order for .cdsenv" in the *Virtuoso Studio Design Environment User Guide.*
CDS_LOG_TIMESTAMPS	Set to `1`, `True`, `Yes`, `t`, or `y` for default timestamp logging This is same as the command line option `-logtime`. Set to `2` or `rel` or `relative` for relative timestamp logging. This is same as the command line option `-logtimerel`. **Note:** The options are case insensitive.

© 2023 All Rights Reserved.

Cadence Environment Variables, *continued*

Variable	What You Can Specify
CDS_USE_LOCAL_TIMESTAMP	If local timestamp is defined and the value does not start with `0`, `f`, or `n`, all timestamps in the `CDS.log` file will use local time instead of Universal Time (UTC) (also known as GMT). **Note:** The options are case insensitive.
CDS_FILTER_CDSENV_WARNINGS	If defined and the value starts with `t` or `y`, warnings for unregistered and type mismatched variables will not be thrown while loading values from the `.cdsenv` file.
CDS_PROMPT_CKOUT	Whether the Auto Checkout form automatically appears when you open data files using a product that has a graphical user interface (GUI) and automatic check-out capability using one of the following keywords: `all` prompts for all *autocheckout* requests and is the default. Causes all tools that have a GUI and perform *autocheckout* operations to prompt you to check out data before performing an *autocheckout*. A prompt is issued regardless of the values of the other variables controlling check in/out behavior. This action lets you set up an environment that prompts for checkout wherever possible. Tools that cannot prompt cannot automatically check out data unless you also set the `CDS_AUTO_CKOUT` variable. ■ `none` does not prompt for any *autocheckout* requests. ■ `views` prompts for *autocheckout* requests of cellview data. ■ `files` prompts for *autocheckout* requests of non-cellview data.

© 2023 All Rights Reserved.

Cadence Environment Variables, *continued*

Variable	What You Can Specify
`CDS_AUTO_CKOUT`	Whether the software automatically checks out a file when you open it with a product using one of the following keywords: ■ `all` checks out all *autocheckout* requests and is the default. ■ `none` does not check out any *autocheckout* requests. ■ `views` checks out *autocheckout* requests of cellview data. ■ `files` checks out *autocheckout* requests of non-cellview data. If a tool has a GUI and `CDS_PROMPT_CKOUT` is set to `all`, `CDS_AUTO_CKOUT` has no effect. Otherwise, when set to `all`, it causes all tools that have an *autocheckout* capability to automatically check out a file as needed.

© 2023 All Rights Reserved.

Cadence Environment Variables, *continued*

Variable	What You Can Specify
CDS_PROMPT_CKIN	Whether the Auto Checkin form appears when you close properties or files that were automatically checked out. Or if you try to exit a session without closing properties or files that were automatically checked out, while using a Cadence product that has both a GUI and automatic check-in capability using one of the following keywords: ■ files prompts you before performing an *autocheckin* of non-cellview data (default). It overrides CDS_AUTO_CKIN and applies to interactive tools only. all prompts for all *autocheckin* requests. Causes all tools that have a GUI and perform check in operations to prompt you whether or not to check in data before performing an *autocheckin*. Tools that do not have a GUI are not affected by this variable. ■ none does not prompt for any *autocheckin* requests. ■ views prompts for *autocheckin* requests of cellview data.
CDS_AUTO_CKIN	Whether the software automatically checks in data files when you close properties or files that were automatically checked out or when you exit a session without closing properties or files that were automatically checked out using one of the following keywords: ■ files file is *autocheckin* if it was automatically checked out during this session and is the default. ■ all checks in all *autocheckin* requests. ■ none does not check in any *autocheckin* requests. ■ views checks in *autocheckin* requests of cellview data. If the tool has a GUI and CDS_PROMPT_CKIN is set to all, CDS_AUTO_CKIN has no effect. When this variable is set to all, all tools automatically check in any file they automatically checked out.

© 2023 All Rights Reserved.

Cadence Environment Variables, *continued*

Variable	What You Can Specify
CDS_IBM_FULLDUMP	Whether to dump the core with full traceback and data sections (IBM only). When an IBM workstation dumps its core, the core contains a full traceback section but no data section. Debugging typically requires the data section.
CDS_LICENSE_DIR	Alternate SoftShare license directory used.
CDS_LOG_VERSION	Naming convention for preserving multiple log files using one of the following keywords: ■ sequential adds a sequential number to the name of the log file, such as CDS.log.1 or CDS.log.2. ■ pid adds the number of the UNIX process to the name of the log file, such as CDS.log.1719 or CDS.log.2250. **Note:** If you do not specify CDS_LOG_VERSION, each session overwrites the log file from the previous one.
CDS_SHM_ADDR	The address to use for shared memory. **Note:** Set this variable only when you do not have shared memory. Setting this variable turns off VO fasttime. If you set CDS_SHM_ADDR to a non-zero hex number, the software uses that address instead of the default address for shared memory.
CDS_AUTO_64BIT	Which applications to run using the 64-bit version All IC* products are 64-bit.

Enable Access to Remote Hosts

Work with your IT team to have the right setup working following all security guidelines.

© 2023 All Rights Reserved.

Modifying the .Xdefaults or Equivalent File

The <*your_install_dir*>/tools.<*plat*>/dfII/cdsuser directory contains a `.Xdefaults` file. The X Window System normally uses the resources in the X resources database that is loaded upon login from your local `.Xdefaults` file.

You can copy the Cadence-provided `.Xdefaults` file to your local directory and modify it to contain the settings you want.

1. Copy Cadence `.Xdefaults` file to your local directory. For example:

   ```
   cd
   cp <your_install_dir>/tools.<plat>/dfII/cdsuser/.Xdefaults .
   ```

2. Modify the settings you want and save the file.

The software reads in X resources when you start your application. If you make changes to your `.Xdefaults` file after the software is running, these changes do not take effect until you read the resources into the X server and restart your application. You must force the server and window manager to see the file changes.

1. Read in the `.Xdefaults` file.

   ```
   xrdb -merge .Xdefaults
   ```

   Note: The `-merge` option prevents deleting desired settings that are not in the `.Xdefaults` file.

2. Log out and login again.

The following table identifies some of the resources that you can modify in your local `.Xdefaults` file. You can find examples of these resource settings in <*your_install_dir*>/tools.<*plat*>/dfII/cdsuser/.Xdefaults.

These resources use tight bindings with a period (`.`) instead of the usual loose bindings with an asterisk (`*`). The period cannot be replaced by an asterisk. Wildcard resources produce unpredictable effects on dialog boxes and menus. Cadence recommends that you restrict your resources to the list shown in the resource list above to prevent X resources from clashing with SKILL descriptions for forms and menus.

Resource	Description
`Opus.activeBannerColor`	Background color of the window number label displayed at the bottom-left corner of the current window.
`Opus.beepDialog`	Beeps when warning or dialog boxes appear.

© 2023 All Rights Reserved.

Resource	Description
`Opus.formPlacement`	Location of form (`top`, `bottom`, `left`, `right`, or `center`).
`Opus.formRelativeTo`	Location of form relative to screen, window (currentWindow), or CIW.
`Opus.geometry`	Location and size of CIW (overrides `Opus.x`, `Opus.y`, `Opus.height`, and `opus.width`).
`Opus.height`	Height of CIW (`Opus.geometry` overrides this setting).
`Opus.optionFormPlacement`	Placement of option form (*top*, *bottom*, l*eft*, *right*, or *center*).
`Opus.optionFormRelativeTo`	Placement of option form relative to *screen*, current window (currentWindow), or CIW.
`Opus.showScrollBars`	Sets the initial value of the `ui` cdsenv variable `showScrollBars`. If set, this will be overridden by the value of `ui.showScrollBars` in the `.cdsenv` file. The value should be set to `True` or `False`. The default is `False`.
`Opus.width`	Width of the CIW (`Opus.geometry` overrides this setting).
`Opus.x`	Horizontal position (X axis) of the CIW (`Opus.geometry` overrides this setting).
`Opus.y`	Vertical position (Y axis) of the CIW (`Opus.geometry` overrides this setting).

Modifying the .cdsinit File

Before you set up users, decide whether you want them to use the same `.cdsinit` file. The program uses the first `.cdsinit` file that it finds in the following order:

1. *<your_install_dir>*`/tools.`*<plat>*`/dfII/local/.cdsinit`
2. Current directory (`.`)
3. Your home directory (`~`)

You can customize the `.cdsinit` file after you copy it into the current directory or the user home directory. Use the following guidelines when modifying the `.cdsinit` file:

© 2023 All Rights Reserved.

Tip

You can find a sample `.cdsinit` file at <*your_install_dir*>/tools.<*plat*>/dfII/samples/local/cdsinit.

- Group the information in the file according to the application and clearly label the application name using comment lines.
- For each setting, use comment lines to specify what the normal defaults are and to what they might be changed. Specify what you normally use.
- Warn about any settings that lead to undesirable results.
- Set the option to your normal value or to the normal default value.
- Read several files if necessary, such as
  - <*your_install_dir*>/tools.<*plat*>/dfII/local/.cdsinit
  - A project's `./.cdsinit`
  - Each user's personal `~/.cdsinit`

To create a site-specific file, follow these steps:

1. Copy the default file to another directory.

   Type the following to copy the default file to a local directory:

   ```
   cp <your_install_dir>/tools.<plat>/dfII/cdsuser/.cdsinit
   <your_install_dir>/tools.<plat>/dfII/local
   ```

2. Edit the last line of the (system) `.cdsinit` file to read.

   ```
   if(isFile("./.cdsinit")
   then load("./.cdsinit")
   else when(isFile("~/.cdsinit")
            load("~/.cdsinit)
        )
   )
   ```

3. Save and close the file.

This allows you to customize their own `.cdsinit` files. Some customizations may include the following:

- You can specify the library search path for the site-specific libraries. See the *Virtuoso Studio Design Environment User Guide* for more information.
- You can define key bindings.

© 2023 All Rights Reserved.

- You can set up the environment for SKILL programming (see the *Cadence SKILL Language Reference* and the *Cadence SKILL Language User Guide*).
  - Set the log filter to display user and program results in the output area of the CIW:

    ```
    hiSetFilterOptions( t t t t t t t )
    ```
  - Turn the `writeProtect` switch off:

    ```
    sstatus( writeProtect nil )
    ```

    This switch affects only procedures loaded after the software starts running.
  - Turn on the debugger:

    ```
    installDebugger()
    ```
- If you have your own SKILL files, you can include one or more locations in the SKILL search path by adding a `setSkillPath` function with a list of space-separated path strings such as one of the following:

  ```
  setSkillPath(". skill_path")
  setSkillPath(strcat(". <your_install_dir>/etc skill_path"))
  setSkillPath(strcat(". " prependInstallPath("/etc ") "skill_path"))
  ```

Related Topics

hiSetFilterOptions

Verifying Your System Configuration

The system configuration checker (`<`*your_install_dir*`>/tools.<`*plat*`>/bin/checkSysConf`) verifies that the operating system level, system configuration, and patch level of your machines match Cadence software requirements. To run the system configuration checker, do the following:

1. In an xterm window, type the following command to get a list of valid release names:

   ```
   checkSysConf -r
   ```

   Valid release names appear on your screen.
2. Run the system configuration checker using a valid release name as follows:

   ```
   checkSysConf validReleaseName
   ```

   For example,

   ```
   checkSysConf <current_release> | more
   ```

The system configuration checker reports system information (such as *Host Name* and *Hostid*) and verifies whether system requirements are met (such as *MEMORY*, *SWAP*,

© 2023 All Rights Reserved.

DISPLAY, *PACKAGE*, and *PATCH* requirements). Any failures to comply with system requirements for the specified release stream appear at the end of the report. For example, you might need to find out which products require a missing patch by running `checkSysConf` with the `-p` option:

```
checkSysConf <current_release>  -p patch
```

For more information about the system configuration checker, refer to the *Cadence Online Support* web site:
http://support.cadence.com/docs/files/releases/sys_conf_check/welcome.html.

Distributing the User Files

Caution

Do not overwrite existing user files.

➤ Copy the files to the new user home directory.

Include the following default or site-specific files:

`.cshrc` or `.profile`
`.Xdefaults`

Configuring Remote Displays

Normally, only specified workstations can access the X server, but you can change the access by typing one of the following commands:

Important

Contact your system administrator before using this command as it overrides more stringent security schemes.

Command	Description
`xhost +`	All hosts can access the X server.
`xhost +` *`hostname`*	Add *`hostname`* to access list.
`xhost -` *`hostname`*	Remove *`hostname`* from access list.
`xhost -`	Only those workstations listed in the access list, `/etc/X*.hosts`, can access the X server.

© 2023 All Rights Reserved.

Library and Tool Issues

You might want to also consider the following library and tool issues:

- Creating new libraries
- Integrating other tools
- Creating menus for tools (see the *Virtuoso Studio Design Environment User Guide*)
- Locale Settings

dbAccess Command-Line Executable

`dbAccess` is a non-graphical UNIX command-line executable for using the core SKILL functionality and accessing DFII OpenAccess design and technology file data. You can also use this executable to debug Pcells and verify that they run in other environments and have no dependencies on product packages or licenses. `dbAccess` supports `db`, `dd`, `rod`, and `tech` functions. Since it does not have a graphical user interface, it does not support `hi` and `ge` functions.

Main Command Options

`dbAccess <[-load <file-name>] [-cdslib *fileName*]>`

Description

`-load`	Loads and executes the specified SKILL file.
`-cdslib *fileName*`	Reads the specfied filename for library definitions. By default `dbAccess` reads the cds.lib file in the current working directory.

If you do not provide a command-line option, `dbAccess` switches to interactive mode and accepts commands from the keyboard or `stdin`.

In addition, `dbAccess` accepts the Virtuoso command-line options, such as: 0

`-V`	Displays Cadence release version.
`-W`	Displays Cadence release subversion.

© 2023 All Rights Reserved.

However, it ignores options that require a graphical user interface.

Example

Suppose you have a SKILL file, `listLibs.il`, which includes commands to display the libraries specified in the `cds.lib` file. If a `cds.lib` file is not present, the default `cds.lib` from the installation directory is used.

The file contents are as follows:

```
foreach(lib ddGetLibList() printf("%s\n" lib~>name))
```

Now, if you run the `dbAccess` command and loads this SKILL file, the following output is displayed on a terminal window:

```
dbAccess -load listLibs.il
Virtuoso Framework License (111) was checked out successfully. Total checkout
time was 0.64s.
cdsDefTechLib
basic
US_8ths
rfLib
rfExamples
ahdlLib
analogLib
functional
```

The above output is generated using the default `cds.lib` from the installation directory.

Related Topics

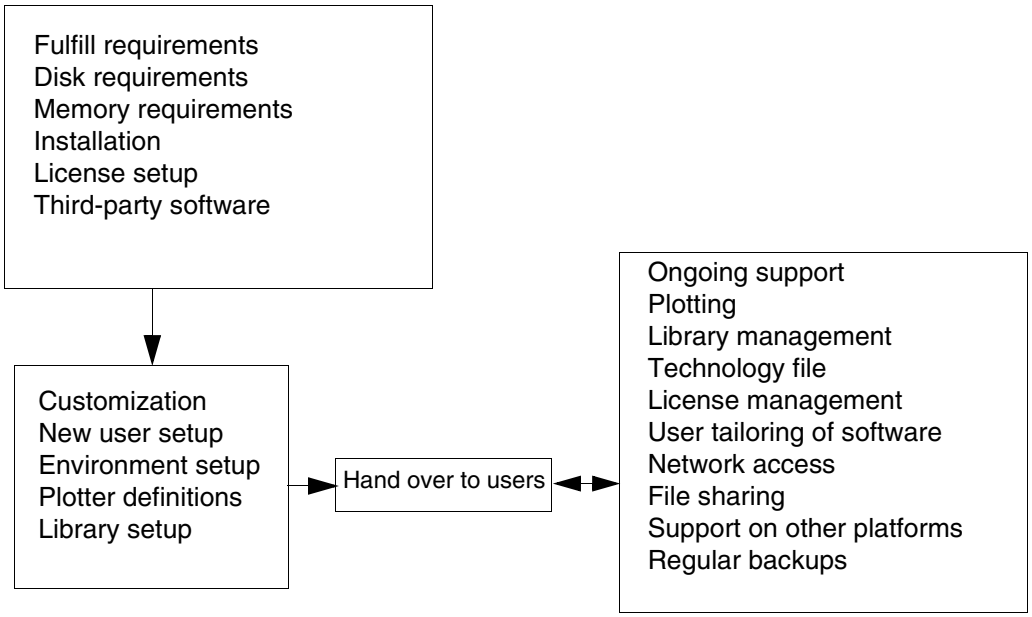
Command-Line Options For Cadence Applications

© 2023 All Rights Reserved.

3

Additional Virtuoso Studio Licensing and Configuration Information

The administration of the software is summarized below.



Note: This document assumes that your operating system and the X Window System are installed and that each user has an account (user name, home directory, `.cshrc`, and `.login` file or `.profile` file). The *Virtuoso Studio Design Environment User Guide* provides more information about customizing these files.

The following information is described in this chapter:

© 2023 All Rights Reserved.

■ Locale Settings on page 98

X Window System

You can find server-dependent directories and files in the following location:

X_install_dir/etc/*

where *X_install_dir* is the directory where your X Window system software is installed.

You can start X in one of the following ways:

1. The workstation runs X all the time (X is running even before you log in). You are using `dtlogin`.

   Note: This document assumes that you are using `dtlogin`.

   Important

   The `dtterm -C` option on some platforms makes the dtterm window act as the console, whereas on other platforms, `dtterm -C` quits with an error.

2. The workstation comes up in text or a proprietary graphics mode. You start X after you log in (without `xdm`).

© 2023 All Rights Reserved.

Running 64-Bit Versions of Applications

Most Cadence® applications have both 32- and 64-bit versions. The 64-bit versions of applications are installed in the same tools hierarchy as the 32-bit versions. A wrapper for each application determines which version of the application is run.

Note: The default setting is to run applications in 32-bit unless the `CDS_AUTO_64BIT` environment variable is set to `ALL` (see below for more information).

To run the 64-bit version of a Cadence application, do the following:

1. Verify that your operating system supports 64-bit applications.

   Note: You should have a minimum of 4GB of RAM to run Cadence 64-bit applications.

2. Verify that a 64-bit version of the application is installed.

   The 64-bit version of an application is located in the `64bit` directory in the standard installation location of the application.

   For example,
   your_install_dir`/tools/bin/64bit/cdsHierEditor`
   your_install_dir`/tools/dfII/bin/64bit/virtuoso`

© 2023 All Rights Reserved.

3. Set the following environment variable:

```
CDS_AUTO_64BIT { ALL | NONE | list | INCLUDE:list | EXCLUDE:list }
```

`ALL`	All applications are run as 64-bit.
`NONE`	All applications are run as 32-bit.
`list`	Only the applications specified are run as 64-bit. Specify *`list`* as a list of case-sensitive application names, separated by a colon, comma, or semi-colon. If you use a semi-colon, enclose the list in quotation marks.
`INCLUDE:`*`list`*	Only the applications specified are run as 64-bit. Specify *`list`* as a list of case-sensitive application names, separated by a colon, comma, or semi-colon. If you use a semi-colon, enclose the list in quotation marks.
`EXCLUDE:`*`list`*	Only the applications specified are run as 32-bit; all other applications are run as 64-bit. Specify *`list`* as a list of case-sensitive application names, separated by a colon, comma, or semi-colon. If you use a semi-colon, enclose the list in quotation marks.

For example, if you have `virtuoso` and `libManager` installed:

`CDS_AUTO_64BIT` is ...	**The following versions are run ...**
`ALL`	`virtuoso: 64-bit; libManager: 64-bit`
`virtuoso`	`virtuoso: 64-bit; libManager: 32-bit`
`INCLUDE:virtuoso`	`virtuoso: 64-bit; libManager: 32-bit`
`EXCLUDE:virtuoso`	`virtuoso: 32-bit; libManager: 64-bit`
`NONE`	`virtuoso: 32-bit; libManager: 32-bit`

Important

Setting `CDS_AUTO_64BIT` does not guarantee that you will run the 64-bit version of an application. The wrapper runs the 64-bit version of the application only if all the following conditions are true:

© 2023 All Rights Reserved.

- The operating system supports 64-bit applications.
- A 64-bit version of the application is installed.
- You choose to run the 64-bit version by setting the `CDS_AUTO_64BIT` environment variable.

  Note: Applications can override the `CDS_AUTO_64BIT` variable with an application-specific variable. See the application's documentation for more information.

Otherwise, the 32-bit version of the application is run.

4. Start the Cadence application from its standard location.

   For example, for the virtuoso executable,
   your_install_dir`/tools/dfII/bin/`*virtuoso*.

   or for the Cadence Hierarchy Editor

   your_install_dir`/tools/bin/64bit/cdsHierEditor`

   The wrapper for the application assess whether to run the 32- or 64-bit version of the application based on the value of `CDS_AUTO_64BIT`.

   The corresponding versions of the applications reside in subdirectories of the `bin` directory. For example, the 32-bit version of *appName* is in *your_install_dir*`/tools/bin/32bit/`*appName*.

Caution

Do not run the executables in the `32bit` or `64bit` directories directly (always run the application through its wrapper).

Note: See your application documentation for more information. In some cases, you might need to start *appName.exe* instead of *appName* to run the application through its wrapper.

Tip

When you start an application, you can use the `-debug3264` option to assess whether you are running the 32- or 64-bit version. Diagnostic information is displayed before the program starts.

In the following example, the 32-bit version of the libManager executable was run even though `CDS_AUTO_64BIT` was set to `ALL`. The output of the `-debug3264` option

© 2023 All Rights Reserved.

indicates that the 32-bit version was run because the 64-bit version of the application was not installed.

```
% setenv CDS_AUTO_64BIT ALL
% libManager.exe -debug3264
-------- 32/64 bit wrapper diagnostics -------------
App name: libManager.exe
App path: /net/machine/cds/5.0.0/tools/dfII/bin
OS is 64-bit capable.
The user has selected 64-bit operation via the environment variables.
No 64-bit version of the application exists.
A 32-bit version of the application exists.
----------------------------------------------------
PATH            :
/net/ansbk/usr/SoftWindows.solaris/bin:/mnt3/ns/bin:/usr/bin/X11:/usr/openwin
/bin:.:/usr/ucb:/bin:/mnt3/ns/bin:/usr/local/bin:/usr/bin:/usr/etc:/usr/local
:/usr/lang/v3:/net/machine/cds/5.0.0/tools/bin:/net/machine/cds/5.0.0/tools/d
fII/bin:/net/machine/cds/5.0.0/tools/lib:/opt/SUNWdtpcv/bin:/usr/local/pvt
LD_LIBRARY_PATH :
/net/machine/cds/5.0.0/tools/lib:/usr/lib:/usr/openwin/lib:/usr/lib/X11:/usr/
dt/lib
CDS_AUTO_64BIT  : ALL
Launching "/net/machine/cds/5.0.0/tools/dfII/bin/32bit/libManager.exe"
----------------------------------------------------
```

TrueColor Visuals

The software defaults to a 24-planes TrueColor visual. If a 24-planes visual is not available, the software will first search for a 16-planes TrueColor visual, then a 15-planes TrueColor visual.

Note: One of these graphics display visuals must be available in order to run TrueColor visuals.

Finding Available Visuals

To check for available graphics display visuals, do the following.

➤ At the system prompt, type `xdpyinfo`

The following shows the results of `xdpyinfo` for a typical 24-planes TrueColor visual:

```
visual:
visual id:    0x2e
class:    TrueColor
```


© 2023 All Rights Reserved.

```
depth:    24 planes
available colormap entries:    256 per subfield
red, green, blue masks:    0xff, 0xff00, 0xff0000
significant bits in color specification:    8 bits
```

Pseudocolor and TrueColor Visuals

In previous releases, the software supported Pseudocolor visuals. Pseudocolor visuals allowed color mapping (mapping between the pixels used to define the colors and the colors that display). In the IC 6.1 release, only TrueColor visuals are supported. TrueColor provide a full palette of colors which increase with larger numbers of planes (bits). In other words, the incremental difference between each color is smaller with the more planes you have.

A 15-plane TrueColor visual provides 32,768 colors and uses 5 bits each to describe the red component, green component, and blue components. A 16-plane TrueColor visual provides 65,535 colors and uses 6 bits to describe the red component, 5 bits to describe the green component, and 5 bits to describe the blue component. The 24-plane TrueColor visual provides 16,777,216 colors and uses 8 bits to describe each component of the red, green, and blue.

© 2023 All Rights Reserved.

The following table shows how the total number of planes (or bits) are used to describe each of the components; red, green, and blue for TrueColor visuals.

15 Planes Color Depth

Color	r	r	r	r	r	g	g	g	g	g	b	b	b	b	b
Bits	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

16 Planes Color Depth

Color	r	r	r	r	r	r	g	g	g	g	g	b	b	b	b	b
Bits	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

24 Planes Color Depth

Color	r	r	r	r	r	r	r	r	g	g	g	g	g	g	g	g	b	b	b	b	b	b	b	b
Bits	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Locale Settings

Virtuoso software requires locale settings to be set to `C` in order to work properly. When you run Virtuoso software on any platform, the software sets the locale to `C` automatically. Your original locale settings are ignored.

This also means that for a script, shell, or other tool that is run from a Virtuoso application (for example, a shell invoked from the CIW), the locale setting that is used is always `C`, regardless of any locale variables set in your environment.

If you want to restore your original locale settings for a script, shell, or tool that is invoked from a Virtuoso application, source one of the following scripts first:

For Bourne shells:

your_install_dir`/tools/dfII/samples/local/cdsOrigLocale.sh`

For C shells:

© 2023 All Rights Reserved.

your_install_dir/tools/dfII/samples/local/cdsOrigLocale.csh

For example,

source *your_install_dir*/tools/dfII/samples/local/cdsOrigLocale.csh

or

. *your_install_dir*/tools/dfII/samples/local/cdsOrigLocale.sh

You can also do this from the `system` command. For example, to run dtpad, you could use the following command:

```
system( strcat( ". " prependInstallPath( "samples/local/cdsOrigLocale.sh" )
            "; /usr/dt/bin/dtpad -standAlone" ) )
```


© 2023 All Rights Reserved.

© 2023 All Rights Reserved.

A

Licensing Environment Variables

This appendix describes the licensing variables that you can use to customize the default license settings. You can set the values of these environment variables in the `.cdsenv` file.

The following example illustrates an environment variable setting in the `.cdsenv` file.

```
license VLSLicenseCheckoutOrder string "XL, GXL, EXL, MXL"
```

In each entry, the first column is the tool partition, the second column is the variable, the third column is the data type, and the fourth column contains the value to be used. For more information, see *Specifying Environment Settings* in *Virtuoso Studio Design Environment User Guide*.

You can also set and retrieve the values of these environment variables using the `envSetVal` and `envGetVal` functions in CIW. For example:

```
envSetVal("license" "VLSLicenseCheckoutOrder" 'string "XL, GXL, EXL, MXL")
envGetVal("license" "VLSLicenseCheckoutOrder")
```

This section describes the environment variables available in the `license` partition of the `.cdsenv` file:

- CheckoutOrder Variables
- UseNextLicense Variables
- Other Licensing VariablesskipAMSEnvironmentLicCheck

© 2023 All Rights Reserved.

CheckoutOrder Variables

These variables control the license checkout order of Virtuoso Studio tier-based applications. The values specified in these variables govern the order in which higher-tier licenses are checked out when an attempt to check out a lower-tier license fails.

For each `CheckoutOrder` variable, a set of valid values can be created using a combination of values from the default value list, with commas or spaces as delimiters. However, an empty string is not allowed as a valid value for these variables, except `EADLicenseCheckoutOrder`.

The following `CheckoutOrder` variables available in the `license` partition:

VLSLicenseCheckoutOrder

VSELicenseCheckoutOrder

VLSAdvOptLicenseCheckoutOrder

maestroCheckoutOrder

EADdatasetCheckoutOrder

AMSEnvLicenseCheckoutOrder

VIVALicenseCheckoutOrder

cdlNetlistCheckoutOrder

© 2023 All Rights Reserved.

VLSLicenseCheckoutOrder

```
license VLSLicenseCheckoutOrder string "XL, GXL, EXL, MXL"
```

Description

Specifies the license checkout order preference for the Virtuoso Layout Suite (VLS) product family.

The default is `XL,GXL,EXL, MXL`.

GUI Equivalent

Command:	*CIW – Options – License – Ordering*
Field:	*Layout*

Examples

```
envGetVal("license" "VLSLicenseCheckoutOrder")
envSetVal("license" "VLSLicenseCheckoutOrder" 'string "EXL, MXL, XL")
```

Related Topics

Software Product License Management Form: Ordering Tab

© 2023 All Rights Reserved.

VSELicenseCheckoutOrder

```
license VSELicenseCheckoutOrder string "L, XL"
```

Description

Specifies the license checkout order preference for the Virtuoso Schematic Editor (VSE) product family.

The default is: `L,XL`

GUI Equivalent

Command:	*CIW – Options – License – Ordering*
Field:	*Schematic*

Examples

```
envGetVal("license" "VSELicenseCheckoutOrder")
envSetVal("license" "VSELicenseCheckoutOrder" 'string "XL, L")
```

Related Topics

Software Product License Management Form: Ordering Tab

© 2023 All Rights Reserved.

VLSAdvOptLicenseCheckoutOrder

```
license VLSAdvOptLicenseCheckoutOrder string "95513" "95511, 95512"
```

Description

Specifies the license checkout order preference for Virtuoso advanced node features.

The default is: `95513, 95511,95512`

GUI Equivalent

Command:	*CIW – Options – License – Ordering*
Field:	*Layout Option*

Examples

```
envGetVal("license" "VLSAdvOptLicenseCheckoutOrder")
envSetVal("license" "VLSAdvOptLicenseCheckoutOrder" 'string "95513, 95511")
```

Related Topics

Software Product License Management Form: Ordering Tab

© 2023 All Rights Reserved.

maestroCheckoutOrder

```
license maestroCheckoutOrder string "Explorer, Assembler"
```

Description

Specifies the license checkout order preference for Maestro (Virtuoso ADE Explorer or Virtuoso ADE Assembler) licenses.

The default is: `Explorer, Assembler`

GUI Equivalent

Command:	*CIW – Options – License – Ordering*
Field:	*Maestro*

Examples

```
envGetVal("license" "maestroCheckoutOrder")
envSetVal("license" "maestroCheckoutOrder" 'string "Assembler, Explorer")
```

Related Topics

Software Product License Management Form: Ordering Tab

© 2023 All Rights Reserved.

EADdatasetCheckoutOrder

```
license EADdatasetCheckoutOrder string "EXL, VVAO"
```

Description

Specifies the license checkout order preference for the Virtuoso Electrically Aware Design license.

The default is: `EXL, VVAO`

GUI Equivalent

Command:	*CIW – Options – License – Ordering*
Field:	*EAD_DATASET*

Examples

```
envGetVal("license" "EADdatasetCheckoutOrder")
envSetVal("license" "EADdatasetCheckoutOrder" 'string "VVAO, EXL")
```

Related Topics

Software Product License Management Form: Ordering Tab

© 2023 All Rights Reserved.

AMSEnvLicenseCheckoutOrder

```
license AMSEnvLicenseCheckoutOrder string "AMS, ADE"
```

Description

Specifies the license checkout order to be considered while running the *Check and Save* feature of Text Editor for verilogams and SystemVerilog views. By default, this feature first looks for the `AMS_Environment` license (Product number `70000`), when the `AMS_Environment` license is not available orthe `Maestro` (Virtuoso ADE Explorer or Virtuoso ADE Assembler) license is checked out.

The default is: `AMS, ADE`

`AMSEnvLicenseCheckoutOrder` replaces the `skipAMSEnvironmentLicCheck`, which will be removed from a future release. Currently, when `skipAMSEnvironmentLicCheck` is set to `t`, AMS is removed from the checkout order.

GUI Equivalent

Command:	*CIW – Options – License – Ordering*
Field:	*AMS_env*

Examples

```
envGetVal("license" "AMSEnvLicenseCheckoutOrder")
envSetVal("license" "AMSEnvLicenseCheckoutOrder" 'string "ADE, AMS")
```

Related Topics

Software Product License Management Form: Ordering Tab

© 2023 All Rights Reserved.

VIVALicenseCheckoutOrder

```
license VIVALicenseCheckoutOrder string "VIVA, ADE,Verifier"
```

Description

Specifies the license checkout order preference for Virtuoso Visualization and Analysis XL. By default, this variable is set to `VIVA,ADE,Verifier`.

If the VIVA license is not available, Maestro (Virtuoso ADE Explorer or Virtuoso ADE Assembler) license is checked out. If both `VIVA` and `ADE` licenses are not available, the `Verifier` license is checked out.

You must set this environment variable in `.cdsinit` or `.cdsenv` file. Do not set this environment variable in the CIW.

The default is: `VIVA,ADE,Verifier`

GUI Equivalent

Command:	*CIW – Options – License – Ordering*
Field:	*VIVA*

Examples

```
envGetVal("license" "VIVALicenseCheckoutOrder")
envSetVal("license" "VIVALicenseCheckoutOrder" 'string "VIVA, Verifier")
```

Related Topics

Software Product License Management Form: Ordering Tab

© 2023 All Rights Reserved.

cdlNetlistCheckoutOrder

```
license cdlNetlistCheckoutOrder string "Schematic, SE"
```

Description

Specifies the license checkout order preference for auCdl netlisting. By default, this variable is set to `Schematic, SE`. If the `Schematic` license (`Virtuoso_Schematic_Editor_L` or `Virtuoso_Schematic_Editor_XL)` is not available, the `SE` license (`Virtuoso_Simulation_Environment`) is checked out.

You must set this environment variable in `.cdsinit` or `.cdsenv` file. Do not set this environment variable in the CIW.

The default is: `Schematic, SE`

GUI Equivalent

None

Examples

```
envGetVal("license" "cdlNetlistCheckoutOrder")
envSetVal("license" "cdlNetlistCheckoutOrder" 'string "SE, Schematic")
```

Related Topics

Software Product License Management Form: Ordering Tab

© 2023 All Rights Reserved.

UseNextLicense Variables

If a license for a requested application is not available, the Use Next License dialog box is displayed, which lets you check out a higher tier license, subject to availability.

These variables control whether to check out the next available license for Virtuoso tier-based applications, when the license required for a requested application is not available.

Depending on the value of this variable, you are given an opportunity to check out a higher tier license, subject to availability.

These environment variables can have the following values:

Value	Use
prompt	Prompts for confirmation before checking out the next tier license. **Note**: During a Virtuoso Studio session, the Next License dialog appears only once for each application. When `prompt` is set as the current value, the Next License dialog is displayed asking you to confirm before checking out the next tier license.
always	Always tries to check out the next tier license when the requested license is not available.
never	Never attempts to check out the next tier license and displays an error message instead.

If the value of these variables is set as an empty string " " (except for `EADLicenseCheckoutOrder`) or has an incorrect variable value, the license tool prints an appropriate CIW warning message about the specific issue. In both cases, the previous checkout order is restored.

If Virtuoso Studio is currently being run in the `-nograph` mode, the license order mechanism will not display the Use Next License dialog if a specific license is not available, and the value of `*_UseNextLicense` is set to `prompt`. Instead, a warning message will be added to the log file.

The following `UseNextLicense` variables available in the `license` partition:

VLSXL_UseNextLicense

© 2023 All Rights Reserved.

VSEL_UseNextLicense

VLSAdvStd_UseNextLicense

VLSAdvOpt_UseNextLicense

VIVA_UseNextLicense

ADE_UseNextLicense

Explorer_UseNextLicense

© 2023 All Rights Reserved.

VLSXL_UseNextLicense

```
license VLSXL_UseNextLicense string "prompt"
```

Description

Sets the `UseNextLicense` value that controls whether to check out the next available license (`Virtuoso_Layout_Suite_GXL`) for Virtuoso Layout Suite XL or not.

Valid values are `prompt, always, never`. The default is `prompt`

GUI Equivalent

Command:	*CIW – Options – License – Ordering – Layout – XL*
Field:	*UseNext*

Examples

```
envGetVal("license" "VLSXL_UseNextLicense")
envSetVal("license" "VLSXL_UseNextLicense" 'string "always")
```

Related Topics

Software Product License Management Form: Ordering Tab

```
license VIVALicenseCheckoutOrder string "VIVA, ADE,Verifier"
license VLSXL_UseNextLicense string "prompt"
```


© 2023 All Rights Reserved.

VSEL_UseNextLicense

```
license VSEL_UseNextLicense string "prompt"
```

Description

Sets the `UseNextLicense` value that controls whether to check out the next available license (`Virtuoso_Schematic_Editor_XL`) for Virtuoso Schematic Editor L or not.

Valid values are `prompt, always, never`. The default is `prompt`

GUI Equivalent

Command:	*CIW – Options – License – Ordering – Schematic – L*
Field:	*UseNext*

Examples

```
envGetVal("license" "VSEL_UseNextLicense ")
envSetVal("license" "VSEL_UseNextLicense " 'string "always")
```

Related Topics

Software Product License Management Form: Ordering Tab

© 2023 All Rights Reserved.

VLSAdvStd_UseNextLicense

```
license VLSAdvStd_UseNextLicense string "prompt"
```

Description

Sets the `UseNextLicense` value that controls whether to check out the next available license (95512) for Virtuoso advanced node applications or not.

Valid values are `prompt, always, never`. The default is `prompt`

GUI Equivalent

Command:	*CIW – Options – License – Ordering – Layout Option – 95512*
Field:	*UseNext*

Examples

```
envGetVal("license" "VLSAdvStd_UseNextLicense)
envSetVal("license" "VLSAdvStd_UseNextLicense" 'string "always")
```

Related Topics

Software Product License Management Form: Ordering Tab

© 2023 All Rights Reserved.

VLSAdvOpt_UseNextLicense

```
license VLSAdvStd_UseNextLicense string "prompt"
```

Description

Sets the `UseNextLicense` value that controls whether to check out the next available license (95511) for Virtuoso advanced node applications or not.

Valid values are `prompt, always, never`. The default is `prompt`

GUI Equivalent

Command:	*CIW – Options – License – Ordering – Layout Option – 9511*
Field:	*UseNext*

Examples

```
envGetVal("license" "VLSAdvOpt_UseNextLicense)
envSetVal("license" "VLSAdvOpt_UseNextLicense" 'string "always")
```

Related Topics

Software Product License Management Form: Ordering Tab

© 2023 All Rights Reserved.

VIVA_UseNextLicense

```
license VIVA_UseNextLicense string "prompt"
```

Description

Sets the `UseNextLicense` value that controls whether to check out the next available license (Maestro), as per the specified checkout order for Virtuoso Visualization and Analysis XL or not.

Valid values are `prompt, always, never`. The default is `prompt`

GUI Equivalent

Command:	*CIW – Options – License – Ordering – VIVA*
Field:	*UseNext*

Examples

```
envGetVal("license" "VIVA_UseNextLicense)
envSetVal("license" "VIVA_UseNextLicense" 'string "always")
```

Related Topics

Software Product License Management Form: Ordering Tab

© 2023 All Rights Reserved.

ADE_UseNextLicense

```
license ADE_UseNextLicense string "prompt"
```

Description

Sets the `UseNextLicense` value that controls whether to check out the next available license (Maestro: Virtuoso ADE Explorer/Virtuoso ADE Assembler) for Virtuoso Analog Design Environment product tiers or not.

Valid values are `prompt, always, never`. The default is `prompt`

GUI Equivalent

Command:	*CIW – Options – License – Ordering* – Maestro – *Explorer*
Field:	*UseNext*

Examples

```
envGetVal("license" "ADE_UseNextLicense)
envSetVal("license" "ADE_UseNextLicense" 'string "always")
```

Related Topics

Software Product License Management Form: Ordering Tab

adeMaestroCheckoutOrder

© 2023 All Rights Reserved.

Explorer_UseNextLicense

```
license Explorer_UseNextLicense string "prompt"
```

Description

Sets the `UseNextLicense` value that controls whether to check out the next available license (Virtuoso ADE Assembler) for Virtuoso ADE Explorer or not.

Valid values are `prompt, always, never`. The default is `prompt`

GUI Equivalent

Command:	*CIW – Options – License – Ordering* – Maestro – *Explorer*
Field:	*UseNext*

Examples

```
envGetVal("license" "Explorer_UseNextLicense)
envSetVal("license" "Explorer_UseNextLicense" 'string "always")
```

Related Topics

Software Product License Management Form: Ordering Tab

maestroCheckoutOrder

© 2023 All Rights Reserved.

Product Families and the Associated CheckoutOrder and UseNextLicense

For applications in product family tiers (such as, Layout, Schematic, and ADE), the checkout behavior is governed by both `CheckoutOrder` and `UseNextLicense` variable settings. If the license required for an application as per the `CheckoutOrder` is not available, the `UseNextLicense` variable for the product family controls whether the higher-tier license should be checked out or not.

The following table lists the product families and the associated `CheckoutOrder` and `UseNextLicense` variables:

Product Family	CheckoutOrder Variable	UseNextLicense Variable
Layout	`VLSLicenseCheckoutOrder`	`VLSXL_UseNextLicense`
Schematic	`VSELicenseCheckoutOrder`	`VSEL_UseNextLicense`
Virtuoso Advanced Node	`VLSAdvOptLicenseCheckoutOrder`	`VLSAdvOpt_UseNextLicense`
Maestro	`maestroCheckoutOrder`	`Explorer_UseNextLicense`
ADE/Maestro	`adeMaestroCheckoutOrder`	`ADE_UseNextLicense`
VIVA	`VIVALicenseCheckoutOrder`	`VIVA_UseNextLicense`

Other Licensing Variables`skipAMSEnvironmentLicCheck`

```
license skipAMSEnvironmentLicCheck boolean nil
```

Description

Skips `AMS_environment l`icense check and uses ADE licenses instead.

The default is `nil`

This environment variable will be deprecated in a future release. Instead, use `AMSEnvLicenseCheckoutOrder`.

GUI Equivalent

None

© 2023 All Rights Reserved.

Examples

```
envGetVal("license" "skipAMSEnvironmentLicCheck")
envSetVal("license" "skipAMSEnvironmentLicCheck" 'boolean t)
```

Related Topics

© 2023 All Rights Reserved.

© 2023 All Rights Reserved.

B

Virtuoso Studio Licensing and Configuration Forms

This topic lists the Virtuoso® Studio Licensing and Configuration forms.

© 2023 All Rights Reserved.

Software Product License Management Form - Status Tab

Displays the checked out and relevant licenses.

Field	Description
Checked out Licenses	Lists the licenses that are currently checked out and their count.
Number	Specifies the license number. For example, *111*, *95011*, and *95115*.
Product Name	Species the name of the product for which the license is checked out.
In Use	Specifies the number of licenses being used.
Relevant Licenses	Lists the DFII licenses relevant to all of the tools incorporated into the workbench.
Number	Specifies the license numbers of relevant licences. For example, *111*, *95011*, and *95115*.
Product Name	Specifies the name of the products relevant to all of the tools incorporated into the workbench.
Users	Displays the License Use form the displays the license usage information (for example: user logins, host names, display time, number of licenses, and license version) for the selected licenses, independent of the license-checkout status. Information in this form is read from the license servers that are defined in the `CDS_LIC_FILE` and is extracted from more than one license server. If you do not have any licenses for a selected feature in any license file, the License Use form will display total and available licenses as “0”.
Refresh	Updates the *Checked Out Licenses* and *Relevant Licenses* lists to show the current list of products being run, and the respective license count, along with any changes to the available license list since the initial access of the Software Product License Management form.

Related Topics

Software Product License Management Form: Status Tab

Software Product License Management Form

© 2023 All Rights Reserved.

Software Product License Management Form: Token Usage Tab

Displays the number of `Virtuoso_LayoutSuite_GXL` tokens required to run each capability and the respective usage status.

Field	Description
Capability	Specifies the application for which the tokens are needed.
Tokens	Specifies the number of licenses needed to use a capability.
Version	Specifies the Virtuoso Studio release version.
Count	Specifies the number of tokens checked out.

Related Topics

Software Product License Management Form: Token Usage Tab

Software Product License Management Form

© 2023 All Rights Reserved.

Software Product License Management Form: Ordering Tab

Provides an interface for defining the license checkout order of the Virtuoso Studio application tiers.

Field	Description
Current Checkout Order	Displays the list of product families along with their current checkout order and *UseNext* settings.
Family	Lists the names of the product license families. For example, Schematic, Layout, Layout Option, ADE, AMSenv, VIVA, and EAD_DATASET. To change the checkout order of a product family, click and drag the required tier up or down to move it to the desired location.
UseNext	Displays the *UseNext* setting, which controls whether to check out the next available license for Virtuoso tier-based applications when the license required for a requested application is not available. If a tiered application has an associated *UseNext* setting, a drop-down list appears next to its name. If you reorder a tier within the product family, the *UseNext* drop-down moves along with it. The *UseNext* drop-down list provides the following options: ■ *prompt:* Confirm before checking out a higher-tier license. ■ *always*: Always check out the higher-tier license (as per selected checkout order). ■ *never*: Never check out the higher-tier license and display an error message instead.

© 2023 All Rights Reserved.

Field	Description
	When *prompt* is set as the *UseNext* drop-down value and the license for a requested application is not available, the Next License dialog box displays, which lets you check out a higher tier license, subject to availability. Four options are offered: *Session*, *Skip*, *Always*, or *Never*. ■ *Session*: Checks out a higher-tier license, if available, for the current session. If a higher-tier license is unavailable, the Next License dialog box will only be displayed once. However, a new virtuoso session will cause the dialog box to display again. The *Session* option does not override the setting in the local .cdsenv file and updates only the internal or in-memory value of the corresponding `UseNextLicense` .cdsenv variable. Because of this, the *Session* response is applicable for the current session only. However, if you want to restore the Next License dialog box settings in the same session, reset the `UseNextLicense` .cdsenv variable to `prompt` using the `envSetVal` function in the CIW or select *prompt* from the *UseNext* drop-down list. ■ *Skip*: Skips the next license. For example, if L license is not available, and you click *Skip* in the Next License dialog box, the XL license will be skipped and an attempt to check out the GXL license will be made. ■ *Always:* Virtuoso will always try to check out the next license, if the requested license is not available. The Next License dialog box is not displayed again when virtuoso is restarted. ■ *Never*: Virtuoso will never try to check out the next license, if the requested license is not available. The Next License dialog box is not displayed again when virtuoso is restarted. The *Always* and *Never* options are stored in the following environment variables in ~/.cdsenv as soon as they are set.

© 2023 All Rights Reserved.

Field	Description

	For example:
	`license VLSXL_UseNextLicense string "prompt"/"always"/ "never"`
	`license VSEL_UseNextLicense string "prompt"/"always"/ "never"`
	
	If a license is not available warning messages will be displayed in the CIW.
	
	The settings selected in the *Current Checkout Order* pane are reflected in the `.cdsenv` variables *`CheckoutOrder`* and *`UseNextLicense`*.

Available Tiers	Displays all applicable tiers for each product family. However, you can choose to deselect a tier and remove it from the current checkout order list by dragging and moving it to *Available Tiers*.
	If there is only one tier in the family tree, you are not allowed to remove it from the *Current Checkout Order* list by dragging and moving it to *Available Tiers* list, because a family tree cannot be empty.
Tier	List of available and applicable tiers in the right pane.
UseNext	Displays the value of the tier.
Save to File	Saves the selected options in the `.cdsenv` file. You can also edit your preferences in the `.cdsenv` file.

Related Topics

Software Product License Management Form: Ordering Tab

Software Product License Management Form

© 2023 All Rights Reserved.

Software Product License Management Form: Diagnostics Tab

Provides an interface for setting the parameters needed for license-checkout reporting.

Field	Description
The Diagnostics Tab Panes	Specifies information related to available and printer licenses along with their respective token capabilities.
Available License	Lists all available licenses.
Printed License	Displays a list of licenses selected for the detailed license-checkout report. When a license listed in the *Printed licenses* pane is checked out, a warning message will be output to the CIW. You can select an appropriate license name from *Available License* pane and click the right arrow (-->) to move it to the *Printed License*s pane. License feature “111” is printed by default in the *Printed licenses* pane as “111” is the required license for Virtuoso applications.
Available token capabilities	Displays the list of available token capabilities for the selected license type. You can list the token capabilities of a particular license by selecting the license name from the drop-down list box provided above this pane.
Printed token capabilities	Displays a list of token capabilities selected for the detailed license-checkout report. When a token capability listed in the *Printed token capabilities* pane is checked out, a warning message will be output to the CIW. You can select an appropriate token capability from the *Available token capabilities* pane and click the right arrow (-->) to move it to the *Printed token capabilities* pane. The selected printed licenses and token capabilities are used to set the value of the `CDS_LIC_PRINT_FILTER` shell variable.

© 2023 All Rights Reserved.

Field	Description
The Diagnostics Tab Controls	The settings specified for the control options in this pane, update the values of the corresponding shell variables.
Timer interval	Sets the timeout interval (in number of seconds) for license search. SHELL Variable: *CDS_LIC_TIMER_INTERVAL*
Print messages	Prints all license call-related messages. You can choose to display only warning messages, or both information and warning messages in the CIW. SHELL Variable: *CDS_LIC_PRINT_TYPE*
Print checkout: time	Prints the time lapse during a license check-out process (in seconds). SHELL Variable: *CDS_LIC_PRINT_TIME*
Show timeout dialog	Displays the timeout message box when the license-checkout time of a license feature exceeds the specified timer interval. SHELL Variable: *CDS_LIC_TIMEOUT_DIALOG* and *CDS_LIC_TIMER_INTERVAL*
Always print checkout messages	Prints a checkout message when a particular license feature is checked out. SHELL Variable: *CDS_LIC_PRINT_ALWAYS*

Related Topics

Software Product License Management Form: Diagnostics Tab

Software Product License Management Form

© 2023 All Rights Reserved.

Software Product License Management Form: Performance Tests Tab

You can run a diagnostic test from the *Performance Tests* tab to test the license checkout performance on a feature-by-feature basis and across each server. This tab is an interface to the `perf_test` utility.

Field	Description
Number	Specifies the license number. For example, *111*, *95011*, and *95115*.
Product Name	Lists of all the DFII license features that are available for testing. You can select multiple license features for testing.
Type of tests	Specifies the type of test that to perform ■ *Server Load* determines the performance of a license server for a given number of clients (or subprocesses). You can select multiple license features for the server load test. However, the test will be performed for the first license feature only. ■ *Network Latency* determines the license-checkout performance on a feature-by-feature basis and across each server. ■ *Checkout Performance* displays the loop of checkout calls checking out the same feature with additional (upgrade) license for each iteration of the loop. The server maintains one job with a number of licenses over the course of this test. The first license server will be used if there are more than one servers specified in the license path. ■ *Feature Exists Performance* tests if the feature exists on the license server(s) specified in the license path (`CDS_LIC_FILE`).
Loop count	Specifies the loop count for checkout calls in the *Loop Count* field. This value is used for the checkout performance and server load tests; the default is 100.

© 2023 All Rights Reserved.

Field	Description
Sub processes	Specifies the number of sub processes for the server in the *Sub processes* field. This value is used for the server load test; the default value is 50.
Repeat count	Specifies the test repeat count to rerun the tests <count> number of times in the *Repeat count* field.The default value is 1.
Repeat interval	Specifies the test repeat interval in the *Repeat interval* field. Valid only if *Repeat count* is specified. Reruns the tests every <interval> seconds. The default value is 3600 seconds (1hour).
XML output file	Specifies the XML file name in which you want to save the output results in the *XML output file* field.
Output directory	Specifies the directory name where you want to save the output results in the *Output directory* field. If the specified directory does not exist, it is automatically created. When the output directory is specified, the XML file is saved in this directory.
Run tests	Starts the license-checkout performance test. After test completion the PerfTest Output window displays the test results. While the performance test is running, license feature names in the left pane are temporarily disabled. These licenses become available only after the previous test run is complete.

Related Topics

Software Product License Management Form: Performance Tests Tab

Software Product License Management Form

© 2023 All Rights Reserved.