

---

# **Cadence Library Manager User Guide**

**Product Version IC23.1**  
**November 2023**

© 2023 Cadence Design Systems, Inc.

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

---

## 1

<u>Introduction to Library Manager</u>	1
<u>Opening the Library Manager</u>	3
<u>Files and File Extensions</u>	5
<u>C-level Database Access (CDBA) File Extensions</u>	5
<u>OpenAccess File Extensions</u>	6
<u>data.dm File</u>	6
<u>pc.db File</u>	6
<u>master.tag File</u>	7
<u>Controlling the Display of Library Information</u>	8
<u>Viewing Categories and Files in List Mode</u>	8
<u>Viewing Categories and Files in Tree Mode</u>	9
<u>Library Manager Toolbar</u>	10
<u>Thumbnail Images of Cellviews</u>	11
<u>Selecting and Moving Data in the Library Manager</u>	12
<u>Selecting Items in Library Manager</u>	12
<u>Deselecting Items in Library Manager List Boxes</u>	12
<u>Moving Data in List Boxes</u>	13
<u>Viewing and Changing File Permissions</u>	15
<u>Opening a UNIX Window from the Library Manager</u>	17
<u>Exiting the Library Manager</u>	18

## 2

<u>Library Management</u>	19
<u>Opening the Library Path Editor</u>	21
<u>Opening a Cellview</u>	22
<u>Creating a New Cellview</u>	24
<u>Filter and Search Options</u>	26
<u>Using the Filter Combo Box</u>	26
<u>Using the Search Option</u>	30
<u>Refreshing the View and Data</u>	32

# Cadence Library Manager User Guide

---

<u>Viewing the Current Cellview Status</u>	33
<u>Renaming a Library</u>	34
<u>Renaming a Cell</u>	36
<u>Renaming a View</u>	38
<u>Renaming Files</u>	40
<u>Renaming a Reference Library</u>	43
<u>Changing Library Reference</u>	45
<u>Deleting a Library</u>	48
<u>Deleting a Cell</u>	50
<u>Deleting a View</u>	52
<u>Deleting Cells Using Filters</u>	54
<u>Deleting Library or Cell Files</u>	57
<u>Hiding Cells and Showing Hidden Cells</u>	59
<u>Showing Hidden Cells</u>	60
<u>Text Cellviews</u>	61
<u>Editing Library Properties</u>	62
<u>Editing Cell Properties</u>	64
<u>Editing View Properties</u>	66
<u>Adding Properties to a Library, Cell, or View</u>	68
<u>Modifying Properties of a Library, Cell, or View</u>	73
<u>Updating a Managed File</u>	74
<u>Library Display Settings</u>	75
<u>Setting Display Options for Libraries</u>	76
<u>Selecting an Icon for a Library Display Attribute</u>	79
<u>Creating New Library Attributes</u>	82
<u>Setting Attributes on a Library</u>	84
<u>Overriding Customized Library Display Settings</u>	85
<u>Creating and Displaying a Combined Library</u>	87
<u>Creating Combined Libraries</u>	87
<u>Display Combined Libraries</u>	88

## 3

<u>Library Browsing</u>	91
<u>Opening the Library Browser Form</u>	92
<u>Setting the Library Browser Form To Open Automatically</u>	94

# Cadence Library Manager User Guide

---

<u>Selecting a View Using the Library Browser</u>	95
---	----

## 4

<u>Data Copying</u>	97
---------------------	----

<u>Copy Function in the Library Manager</u>	98
---	----

<u>Pre-Copy Checks in the Library Manager</u>	99
---	----

<u>Copying a Library in the Library Manager</u>	101
---	-----

<u>Copying to a New Library</u>	102
---------------------------------	-----

<u>Resolving Copy Problems</u>	104
--------------------------------	-----

<u>Copying a Cell in the Library Manager</u>	106
--	-----

<u>Copying a Cell Hierarchy in the Library Manager</u>	108
--	-----

<u>Updating Cell Instances</u>	110
--------------------------------	-----

<u>Adding a Copied Cell to a Category</u>	111
---	-----

<u>Skip Libraries Together with Update Instances</u>	111
--	-----

<u>Copying a View in the Library Manager</u>	113
--	-----

<u>Copying a View to Another Name or Cell in the Same Library</u>	114
---	-----

<u>Copying a View to Another Library</u>	114
--	-----

<u>Copying a View Hierarchy in the Library Manager</u>	116
--	-----

<u>Updating View Instances</u>	117
--------------------------------	-----

<u>Copying a Library File</u>	119
-------------------------------	-----

<u>Copying a Library File to Another Name in the Same Library</u>	119
---	-----

<u>Copying a Library File to Another Library</u>	120
--	-----

<u>Copying a Cell File in the Library Manager</u>	121
---	-----

<u>Selecting Text in the Copy Wizard</u>	124
--	-----

<u>Selecting Items for Editing in the Copy Wizard</u>	124
---	-----

<u>Deselecting Items in the Copy Wizard</u>	124
---	-----

<u>Selecting Items for Copying in the Copy Wizard</u>	125
---	-----

<u>Editing Text in the Copy Wizard</u>	127
--	-----

<u>Editing a Single Item</u>	127
------------------------------	-----

<u>Editing All Items in a Selected Set</u>	127
--	-----

<u>Editing All Items in a column</u>	128
--------------------------------------	-----

<u>Performing a Simple Copy Using the Copy Wizard</u>	130
---	-----

<u>Copying a Hierarchy Using the Copy Wizard</u>	132
--	-----

<u>Copying an Exact Hierarchy Using the Copy Wizard</u>	135
---	-----

<u>Copying Specific View Using the Copy Wizard</u>	138
--	-----

## Cadence Library Manager User Guide

---

<u>Copying Specific Cells in a Configuration Using the Copy Wizard</u> .....	140
<u>Setting Copy and Rename Preferences</u> .....	142

### 5

## Design Management .....

<u>Setting the Checkin and Checkout Properties</u> .....	146
<u>Canceling Checkout for Properties</u> .....	149
<u>Checking In and Checking Out Categories</u> .....	150
<u>Canceling the Checkout of a Category</u> .....	151
<u>Checking In Files and Properties Automatically</u> .....	152
<u>Controlling Automatic Checkin Behavior</u> .....	153
<u>Always Ask Me</u> .....	153
<u>Never Ask Me</u> .....	153
<u>Auto Checkin Environment Variable Settings</u> .....	155
<u>CDS_PROMPT_CKIN</u> .....	155
<u>CDS_AUTO_CKIN</u> .....	155
<u>Checking Out Files and Properties Automatically</u> .....	157
<u>Controlling Automatic Checkout Behavior</u> .....	158
<u>Always Ask Me</u> .....	158
<u>Never Ask Me</u> .....	158
<u>Auto Checkout Environment Variable Settings</u> .....	160
<u>Submitting Changes to the Design Management System</u> .....	162
<u>Updating an Item Using Design Manager</u> .....	163
<u>Updating Workarea Using Design Manager</u> .....	164
<u>Cellview and File Versions</u> .....	166
<u>Copying a Version of a Cellview or File</u> .....	167
<u>Cellview Version Rules</u> .....	170
<u>Design Management File Status</u> .....	172
<u>Design Management Status Settings</u> .....	172
<u>Displaying the Update Needed Icon</u> .....	172
<u>Check In, Check Out, and Cancel Check Out Using the DM Status Form</u> .....	173

### 6

## Library Creation .....

<u>Creating a New Library in the Library Manager</u> .....	178
--	-----

## Cadence Library Manager User Guide

---

<u>Database Compression Using the oazip Utility</u>	181
<u>Compressing a Library Using Library Manager</u>	184
<u>Compiling an ASCII Technology File</u>	189
<u>Referencing Existing Technology Libraries</u>	190
<u>Attaching a New Library to an Existing Technology Library</u>	191
<u>Creating a New Library Without Specifying a Technology File</u>	192

## 7

<u>Files in Read-Only Mode</u>	193
<u>Getting a List of Locked Cellviews</u>	194
<u>Making Cellviews Read-Only</u>	195

## 8

<u>Category Management</u>	197
<u>Creating a Category Using Library Manager</u>	198
<u>Editing a Category Using Library Manager</u>	200
<u>Deleting a Category Using Library Manager</u>	202
<u>Renaming a Category Using Library Manager</u>	203
<u>Creating a New Category That Includes Subcategories</u>	204
<u>Creating a Subcategory in an Existing Category</u>	205
<u>Modifying a Category to Include a Subcategory</u>	207

## 9

<u>Library Manager Customization</u>	209
<u>Library Manager Customizations Using SKILL Functions</u>	210
<u>About cdsLibMgr.il File</u>	210
<u>Customization of Menus Using the cdsLibMgr.il File</u>	214
<u>Callback Function Triggers</u>	215
<u>Library Manager Customizations in Standalone Mode and with Other Processes</u>	216
<u>Library Manager Customizations in Standalone Mode</u>	216
<u>Library Manager Customization with Other Processes</u>	216
<u>Restrictions on the Library Manager Customization File</u>	217
<u>Actions in the Library Manager Customization File</u>	218
<u>GUI Objects Supported in the Customization File</u>	219

## Cadence Library Manager User Guide

---

<u>Issues with Virtuoso Studio Design Environment SKILL</u>	219
<u>Caution with Pre-Map Callbacks</u>	219
<u>Library Manager Customization Using the .cdsenv File</u>	221
<u>.cdsenv File Search Path Order</u>	221
<u>Using UNIX to Add Settings to a .cdsenv File</u>	222
<u>Saving Settings to a .cdsenv File</u>	223
<u>Loading Settings from a .cdsenv file</u>	225
<u>Using the .libsel File to Customize the Library Manager</u>	227
<u>Settings in the .Xdefaults file</u>	228
<u>Settings in the .libmgr file</u>	230

## 10

### Library Manager Environment Variables

<u>copyValuesFrom</u>	232
<u>useOptionText</u>	233
<u>useOptionsOn</u>	234
<u>selectMatches</u>	235
<u>useOptionText</u>	236
<u>useOptionsOn</u>	237
<u>useOptionText</u>	238
<u>useOptionsOn</u>	239
<u>addToCategoryName</u>	240
<u>addToCategoryOn</u>	241
<u>addToCellsPattern</u>	242
<u>allViewsOn</u>	243
<u>exactHierOn</u>	244
<u>existenceCheck</u>	245
<u>extraViews</u>	246
<u>hierOn</u>	247
<u>rerefCustomVias</u>	248
<u>skipLibsOn</u>	249
<u>updateChoice</u>	250
<u>viewsText</u>	251
<u>addPropFiles</u>	252
<u>overwriteAll</u>	253



## Cadence Library Manager User Guide

---

<u>addCellPropFiles</u>	254
<u>addLibPropFiles</u>	255
<u>expandRadio</u>	256
<u>mpsRadio</u>	257
<u>useMonitor</u>	259
<u>warnRenameDM</u>	260
<u>openView</u>	261
<u>toLibrary</u>	262
<u>toView</u>	263
<u>useOptionText</u>	264
<u>useOptionsOn</u>	265
<u>addToCategoryName</u>	266
<u>addToCategoryOn</u>	267
<u>addToCellsPattern</u>	268
<u>existenceCheck</u>	269
<u>extraViews</u>	270
<u>rerefCustomVias</u>	271
<u>updateChoice</u>	272
<u>updateOn</u>	273
<u>mapTimeout</u>	274
<u>showDFIIWarning</u>	275
<u>startupFile</u>	276
<u>ddDb</u>	277
<u>server</u>	278
<u>fileRadio</u>	279
<u>saveAllOn</u>	280
<u>saveAsText</u>	281
<u>libCheckOn</u>	282
<u>localRadio</u>	283
<u>regExpOn</u>	284
<u>overrideRadio</u>	285
<u>localRadio</u>	286
<u>viewFilterList</u>	287
<u>enableDmQuery</u>	288
<u>showExtendedStates</u>	289
<u>autoModuleNameUpdate</u>	290

## Cadence Library Manager User Guide

---

<u>matchModuleNameCellName</u>	291
<u>sortCellNameCompareFn</u>	292
<u>libSelectCellViewCombineMode</u>	293
<u>skipLibsText</u>	295
<u>dmSyncDelay</u>	296
<u>enableDmSyncDelay</u>	297
<u>showHiddenLibraries</u>	298
<u>showLibraryColors</u>	299
<u>showLibraryCustomIcons</u>	300
<u>showListViewIcons</u>	301
<u>cellFilter</u>	302
<u>viewFilter</u>	303
<u>addHostID</u>	304
<u>addProcID</u>	305
<u>addSequence</u>	306
<u>addUserID</u>	307
<u>baseName</u>	308
<u>categoryText</u>	309
<u>cellLevelText</u>	310
<u>dblClickEditCellView</u>	311
<u>libraryText</u>	312
<u>showCategoriesOn</u>	313
<u>showFilesOn</u>	314
<u>showNonVirtuosoViewtypes</u>	315
<u>viewLevelText</u>	316
<u>dmRadio</u>	317
<u>pathText</u>	318
<u>windowBehavior</u>	319
<u>forceEnv</u>	320
<u>useDMfilter</u>	321
<u>useFastDM</u>	322
<u>updateOn</u>	323
<u>refreshSessionOn</u>	324
<u>useNameOn</u>	325
<u>useNameText</u>	326
<u>useOptionText</u>	327

## Cadence Library Manager User Guide

---

<u>useOptionsOn</u>	328
<u>useNameOn</u>	329
<u>useOptionsOn</u>	330
<u>compressionOn</u>	331
<u>enable</u>	332
<u>display</u>	333
<u>generate</u>	334
<u>verbose</u>	335

## 11

<u>Library Manager Forms</u>	337
<u>Access Permission Form</u>	340
<u>Add Property Form</u>	341
<u>Attach Library to Technology Library Form</u>	342
<u>Cancel Check Out Form</u>	343
<u>Cell Property Editor Form</u>	345
<u>Change Library References Form</u>	347
<u>Check In Form</u>	348
<u>Check Out Form</u>	352
<u>Choose Environment File To Load Form</u>	354
<u>Copy Cell File Form</u>	355
<u>Copy Cell Form</u>	356
<u>Copy Cellview Version Form</u>	358
<u>Copy Library File Form</u>	359
<u>Copy Library Form</u>	360
<u>Copy Preferences Form</u>	361
<u>Copy View Form</u>	363
<u>Copy Wizard Form (Simple Copy)</u>	365
<u>Copy Wizard Form (Hierarchical)</u>	367
<u>Copy Wizard Form (Exact Hierarchy)</u>	369
<u>Copy Wizard Form (By View)</u>	371
<u>Copy Wizard Form (By Configuration)</u>	373
<u>Delete By View Form</u>	375
<u>Delete Cells Form</u>	376
<u>Delete Cell Views Form</u>	377

## Cadence Library Manager User Guide

---

<u>Delete Libraries Form</u>	378
<u>Delete Library Views Form</u>	379
<u>Display Options Form</u>	380
<u>Display Settings Form</u>	383
<u>DM File Status Form</u>	384
<u>New Category Form</u>	386
<u>Library Browser Form</u>	387
<u>Library Manager Form</u>	388
<u>Library Property Editor Form</u>	389
<u>Load Technology File Form</u>	391
<u>Modify 'propertyName' Form</u>	392
<u>New Category Form</u>	393
<u>New File Form</u>	394
<u>New Library Form</u>	395
<u>Reference Existing Technology Libraries Form</u>	396
<u>Rename Cell Form</u>	397
<u>Rename File Form</u>	398
<u>Rename Library Form</u>	399
<u>Rename Reference Library Form</u>	400
<u>Rename View Form</u>	401
<u>Save Library Manager Defaults Form</u>	402
<u>Select an icon Form</u>	403
<u>Submit Form</u>	405
<u>Technology File for New Library Form</u>	406
<u>Version Information Form</u>	407
<u>View Property Editor Form</u>	408

---

# Introduction to Library Manager

---

You can use the Library Manager to create, add, copy, delete, and organize libraries and views in a design project. More specifically, you can

- Import and access design data in libraries under design management control (check out, check in, and version control)
- Define the path to the libraries you want your Cadence design software to access in the `cds.lib` file.

For more information, see the [\*Cadence Library Path Editor Help\*](#).



## Caution

***Virtuoso stops working while launching the Library Manager, if there is an inaccessible path in the `cds.lib`. Virtuoso process would be unresponsive to `mpsImport` requests initiated by the Library Manager. You need to remove or modify the `cds.lib` entry to proceed.***

- Create new libraries in your directories
- Copy data into libraries
- Delete libraries
- Rename libraries, cells, views, files, or reference libraries
- Edit library, cell, and view properties
- Organize cells into categories to help you quickly locate them
- Change permissions for files and views
- Open a UNIX window to locate files and hierarchies
- Customize the colors of the Library Manager user interface using commands in the `.Xdefaults` file
- Navigate libraries, cells, views, files, and categories

## Cadence Library Manager User Guide

### Introduction to Library Manager

---

A record of the commands used during your Library Manager session is stored in the `libManager.log` file in your current working directory.

**Note:** The Library Manager only lets you edit and manage OpenAccess libraries. You need to convert a CDB library to OpenAccess before you can use it with the Library Manager. Contact Cadence Customer Support to discuss how to migrate CDB data to OpenAccess. A CDB library has a library-level `prop.xx` file. Such libraries are grayed-out in the Library Manager and their contents are not displayed. None of the Library Manager commands can be used on the library. For example, you cannot copy the library or delete it. Cell-level `prop.xx` files are also grayed out. Library Manager commands cannot be used on these files. When you copy libraries or cells, the `prop.xx` file is not copied. Copy commands can also fail if you have `prop.xx` files.

If an OpenAccess library erroneously contains a `prop.xx` file, you need to delete the file.

### ***Related Topics***

[Opening the Library Manager](#)

[Controlling the Display of Library Information](#)

[Library Manager Toolbar](#)

[Selecting and Moving Data in the Library Manager](#)

[Viewing and Changing File Permissions](#)

[Opening a UNIX Window from the Library Manager](#)

[Viewing the cds.lib Updates](#)

## Opening the Library Manager

You can open the Library Manager form in standalone mode from an xterm or command tool window or in integrated mode from the Command Interpreter Window (CIW).

To open the Library Manager from an xterm or command tool window:

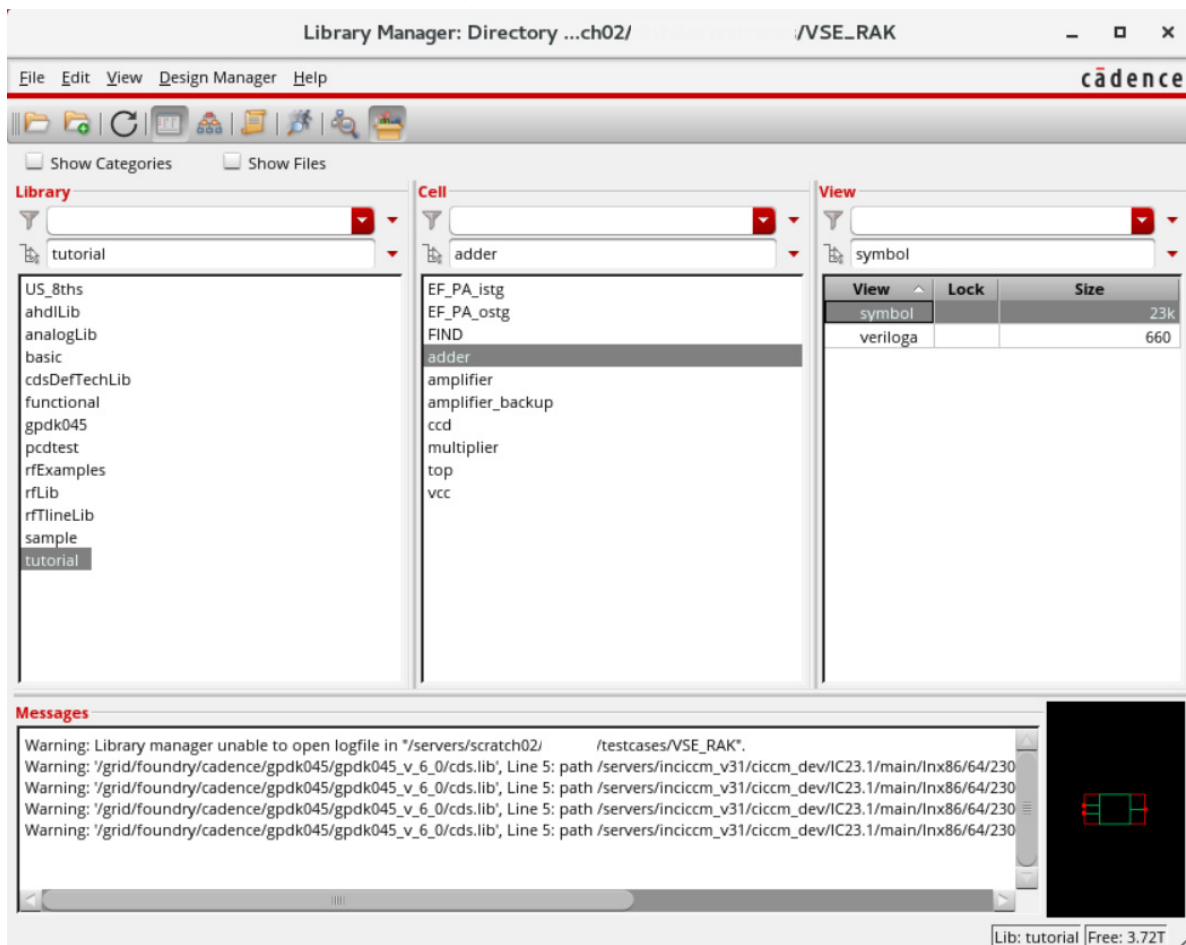
- ➔ Type the command:

```
libManager &
```

The Library Manager appears as a standalone application. In standalone mode, you cannot open cellviews.

To open the Library Manager from the CIW:

1. Choose *Tools – Library Manager*.



## Cadence Library Manager User Guide

### Introduction to Library Manager

---

The applications, such as Library Manager, Library Selector, and Library Path Editor, starts with the same font as Virtuoso. Once you choose the font using the Set Fonts dialog box, the font of these applications starts changing accordingly.

#### ***Related Topics***

[Library Manager Form](#)



## Files and File Extensions



***Do not modify the .cd% and .cd+ files.***

This topic contains information about the various files and file extensions used in Library Manager.

For information about .TopCat and .Cat files, see [Category Management](#).

### C-level Database Access (CDBA) File Extensions

A C-level Database Access (CDBA) file has five possible extensions as follows:

Extension	Description
.cdb	Contains design data, such as schematic, layout, behavioral, or user-customized. For example, Cadence software reads the <code>sch.cdb</code> file when you open a schematic for editing. To open this file using SKILL, use the <code>dbOpenCellViewByType</code> function.
.cd%	Contains cellview information between saves. It is a backup CDBA file.
.cd+	The system maintains this file. It is a temporary auto-save CDBA file.
.cd-	Panic CDBA file Contains panic information in the case of a program crash. You can access this file using the <code>dbOpenPanicCellView</code> SKILL function. It is a panic CDBA file.
.cd?	It is a corrupted CDBA file.

The library directory might also contain a `techfile.cds` file, which is the technology database if the library is a technology library.

## OpenAccess File Extensions

An OpenAccess file has a `.oa` extension. This file contains design data, whether it is schematic, layout, behavioral, or user-customized. For example, Cadence software reads the `sch.oa` file when you open a schematic for editing. To open this file using SKILL, you can use the `dbOpenCellViewByType` function.

The library directory might also contain a `tech.db` file, which is the technology database if the library is a technology library.

## data.dm File

The *property bag* file contains object properties. For OpenAccess 2.2, the file is called `data.dm`.

The contents of this file depend on its location as follows:

Location	Contents
Library directory	Properties that affect the entire library, such as the technology binding
Cell directory	Cell properties, such as CDF descriptions
View directory	Cellview properties

You can use the SKILL functions described under in [Virtuoso Studio Design Environment SKILL Reference](#) to access property bag data.

## pc.db File



Do not modify this file.

The `pc.db` file contains on-disk parent/child relationship information for a cellview, that is, information as of the last `dbSave`, as opposed to what might be in virtual memory.

The following tasks and applications use the `pc.db` file:

- Hierarchical copy
- Hierarchy Editor
- Schematic editor hierarchy traversal function, which many applications use

If `pc.db` file exists, hierarchical copy reads the `pc.db` file to get the parent-child information. Otherwise, hierarchical copy extracts the parent-child information directly from OpenAccess database.

The `pc.db` file allows the traverser to navigate through other non-CDBA data descriptions such as VHDL and Verilog while avoiding the differences in VM CDBA.

Only the super master (`<library>.<default_subcell>`) is physically saved on disk. Therefore, when the copy command is executed using Library Manager, only that subcell is copied to the destination library.

## master.tag File



Do not modify this file.

The `master.tag` file contains master information for a cellview. The master information determines what tool to use when you edit a cellview.

The `master.tag` file might contain any of the following:

- `sch.oa`
- `layout.oa`
- `text.txt`
- `verilog.v`

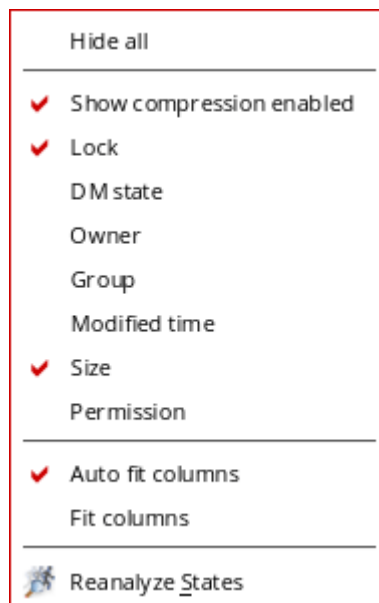
## Related Topics

[Viewing and Changing File Permissions](#)

## Controlling the Display of Library Information

You can choose to display library information from the following options:

- To display information in list boxes, choose *View – Lists* or click the *View Lists* icon from the Library Manager toolbar.
- To display information in tree mode, choose *View – Tree* or click the *View Tree* icon from the Library Manager toolbar.
- To customize the information displayed in the tree table, right click over the table header of the *Libraries* section. The following menu is displayed:



## Viewing Categories and Files in List Mode

To view categories and files in List mode:

1. Select the *Show Categories* check box.

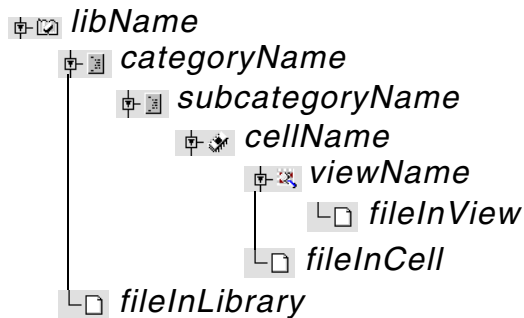
The *Category* list box appears between the *Library* and *Cell* list boxes.

2. Select the *Show Files* check box. The *Files in* list box is displayed under *View* box.

This section shows the *Library* and *Cell* tabs, each of these tabs show the list of files under library and cells respectively.

## Viewing Categories and Files in Tree Mode

Tree mode shows you libraries, cells, views, categories, and files in a hierarchical tree structure:



To view categories and files in Tree mode:

1. Click the arrow to the left of each level to expand that branch of the tree.  
The contents of the expanded branch appear.
2. Categories and subcategories, if there are any, appear at the top levels of the library branch. Files appear at the top of the expanded branches for libraries and cells:
  - ☐ Files in libraries appear beneath either categories, if there are any, or cells if there are no categories.
  - ☐ Files in cells appear beneath views.

### ***Related Topics***

[Category Management](#)

## Library Manager Toolbar

Selecting *View – Toolbar* displays a *Library Manager Toolbar* below the menu bar that provides quick access to a number of useful commands that are also found in the Library Manager menus.



The *Library Manager Toolbar* contains the following short-cut tools:

Fields	Description
<i>Open For Edit</i>	Opens a selected <i>View</i> in the appropriate application for edit.
<i>Open With</i>	Displays the Open File form where you can select what particular application that you want to open a selected view with.
<i>Refresh</i>	Refreshes the Library Manager, design data, and the CDF data.
<i>View Lists</i>	Displays library, cell and view content in a list format.
<i>View Tree</i>	Displays library, cell, and view content in a hierarchical tree format.
<i>Show Categories</i>	Toggles on and off the display of the <i>Categories</i> column list or tree information.
<i>Reanalyze States</i>	Retrieves and displays the latest file states when selecting a new cell in the <i>List</i> view, or an <i>Open/Close</i> of a cell in the tree view. This command works similarly to the <i>View – Refresh</i> menu option but has lower overhead as it does not regenerate the library file contents.  File states are not tracked if this option is selected.
<i>Show open cellviews in use</i>	Reloads the lists in the window with icons next to names to indicate if a library, cell, or view is opened in read mode or edit mode.
<i>Show non-Virtuoso view types</i>	Displays the non-Virtuoso view types for the selected cell.

## Thumbnail Images of Cellviews

Cellview previews are provided in the lower right area of the Library Manager window, in both the tree and list views.

These thumbnail images are an approximate representation of a cellview, with some details that cannot be appropriately displayed removed. Thumbnail cellview previews can however aid selection before opening a view.

You can update displayed thumbnail images by selecting *Edit – Update Thumbnails* or right-clicking in the Library Manager and selecting *Update Thumbnails* from the context-menu presented.

Thumbnail images that represent Pcells can be identified with a Pcell watermark in the lower left corner of the image.

For more information, see [hiGenerateThumbnails](#) in the *Cadence User Interface SKILL Reference*.

### ***Related Topics***

[enable](#)

[display](#)

[generate](#)

[verbose](#)

[Viewing the cds.lib Updates](#)

## Selecting and Moving Data in the Library Manager

The following tasks cover ways you can use the Library Manager forms and list boxes to manipulate data.

### Selecting Items in Library Manager

To select items in Library Manager:

1. Click the item name to select a library, category, cell, view, or file on the Library Manager form whether in List mode or in Tree mode.  
  
**Note:** You cannot select multiple items of the same kind.
2. Right-click the item name to select a library, category, cell, view, or file and display a pop-up menu.
3. If you are in *View – Lists* mode, you can select a specific item not visible in the list box by typing the name in the active field at the top of the list box.

### Deselecting Items in Library Manager List Boxes

To deselect an item, click the item name.

- If you deselect a cell, any selected view is also deselected.
- If you deselect a category, any selected cell and view are also deselected.
- If you deselect a library, any selected item in that library is also deselected.



## Moving Data in List Boxes

Some forms, such as Delete forms and Category forms, use “Do” and “Don’t Do” list boxes. You can move data from one list box to the other to specify a group of items affected by the action.



To move items from one list box to the other:

1. Select the item or items you want to move.

You can select multiple items by Shift-clicking, Control-clicking, clicking and dragging, or using filters.

2. Click the arrow that points in the direction you want to move the items.

The items move from one list box to the other.

### ***Related Topics***

[Controlling the Display of Library Information](#)

[Rename Reference Library Form](#)

# **Cadence Library Manager User Guide**

## Introduction to Library Manager

---

### Delete Libraries Form

## Viewing and Changing File Permissions

If you or your group own a library, cell, view, or file, you can change the permissions to control access. These are the same read, write, execute (*rxw*) permissions you can change from a UNIX command line.

If the software cannot find the item, it generates an error message telling you that the permissions could not be changed. As the owner of an item, you can change the permissions, but you cannot change the ownership.

To view or change the access permissions for an item:

1. On the Library Manager form, select the item from the tree or appropriate list box.
2. Choose *Edit – Access Permissions*.

The Access Permission form appears.

Access Permission			
Library basic			
Owner	datasync		
Group	cadence2		
<b>Access Permission</b>			
Owner	<input checked="" type="checkbox"/> Read	<input checked="" type="checkbox"/> Write	<input checked="" type="checkbox"/> Execute
Group	<input checked="" type="checkbox"/> Read	<input type="checkbox"/> Write	<input checked="" type="checkbox"/> Execute
Others	<input checked="" type="checkbox"/> Read	<input type="checkbox"/> Write	<input checked="" type="checkbox"/> Execute

Apply Close Help

The item you chose appears at the top of the form. The specified owner and group for the selected item appear in the corresponding fields.

If the item you select is under design management and is either checked in or checked out to someone other than you, you cannot select the *Access Permissions* command.

3. Select check boxes for the permission values you want to set.
  - ☐ By default, the owner has read, write, and execute permissions.

## Cadence Library Manager User Guide

### Introduction to Library Manager

---

- ☐ By default, the owner must have write permission before group or others can have write permission.
- ☐ By default, if group or others have write or execute permission, they must also have read permission.

4. Click *Apply*.

The Library Manager changes the permissions for the selected item.

#### ***Related Topics***

[Access Permission Form](#)

## Opening a UNIX Window from the Library Manager

If you want to create or edit a `cds.lib` file, display a directory structure, display information in man pages, or use other UNIX functions, you can open an xterm (UNIX shell) window from the Library Manager.

To open an xterm window:

- ➡ Choose *File – Open Shell Window*.

An xterm window appears. The working directory is the one from which you started either the Library Manager or your design environment application.

### ***Related Topics***

[Library Manager Form](#)

[Viewing the cds.lib Updates](#)


[Creating Library Definitions File](#)

## Exiting the Library Manager

To exit the Library Manager:

- ➡ Choose *File – Exit*.

The Library Manager closes. Your design environment application does not close.

**Note:** Clicking the *Close* button  does not close the Library Manager window, instead it minimizes the window. If needed, you can restore this window from the Status bar.

### ***Related Topics***

[Library Manager Form](#)

---

# Library Management

---

This section discusses various features of Library Manager.

## *Important*

The Library Manager only lets you edit and manage OpenAccess libraries. You need to convert a CDB library to OpenAccess before you can use it with the Library Manager. Contact Cadence Customer Support to discuss how to migrate CDB data to OpenAccess.

A CDB library has a library-level `prop.xx` file. Such libraries are grayed-out in the Library Manager and their contents are not displayed. None of the Library Manager commands can be used on the library. For example, you cannot copy the library or delete it. Cell-level `prop.xx` files are also grayed out. Library Manager commands cannot be used on these files. When you copy libraries or cells, the `prop.xx` file is not copied. Copy commands can also fail if you have `prop.xx` files.

**Note:** If an OpenAccess library erroneously contains a `prop.xx` file, you need to delete the file.

## ***Related Topics***

[Opening the Library Path Editor](#)

[Opening a Cellview](#)

[Filter and Search Options](#)

[Renaming a Library](#)

[Renaming a Reference Library](#)

[Renaming a Reference Library](#)

[Adding Properties to a Library, Cell, or View](#)

## **Cadence Library Manager User Guide**

### **Library Management**

---

#### Modifying Properties of a Library, Cell, or View

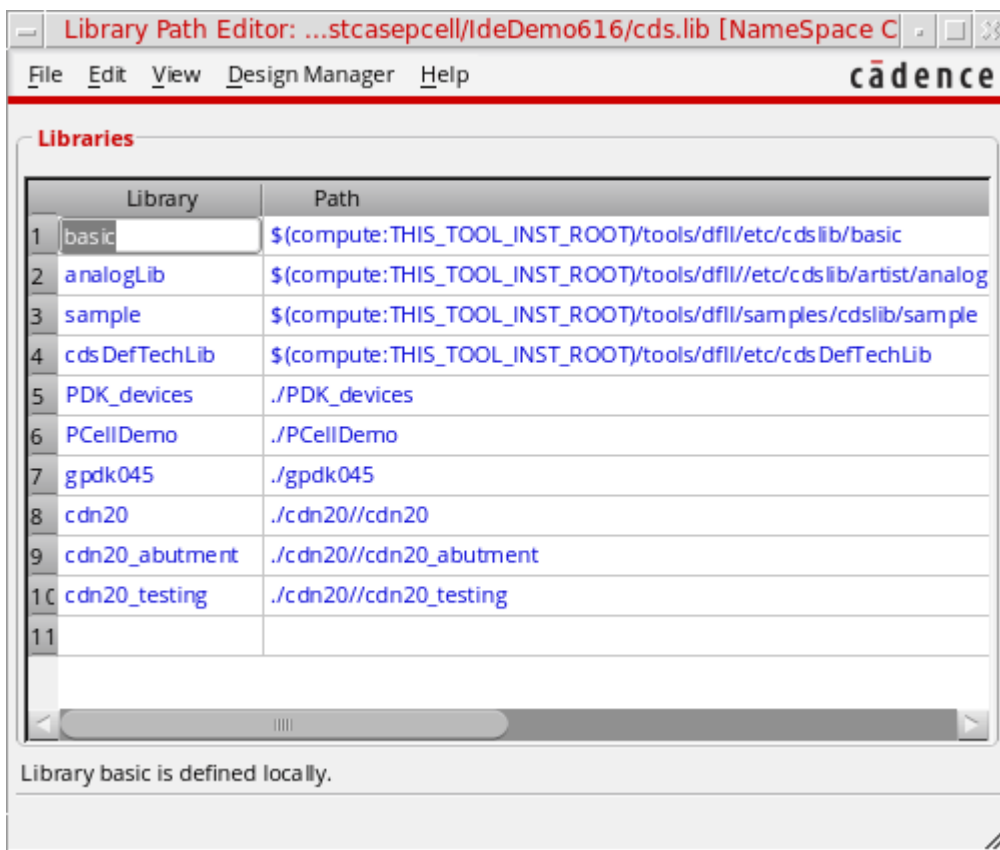


## Opening the Library Path Editor

To open the Library Path Editor:

1. Open the Library Manager.
2. Choose *Edit – Library Path*.

The Library Path Editor form appears.



The Library Path Editor lets you view and edit the information in your `cds.lib` file, which defines the location of the reference and design libraries you want to use in your design.

### ***Related Topics***

[Opening the Library Manager](#)

## Opening a Cellview

To open a cellview:

1. In the CIW, choose *Tools – Library Manager*.

Libraries defined in your `cds.lib` file and the default technology library `cdsDefTechLib` appear on the Library Manager form.

When you open the Library Manager in standalone mode, you cannot open cellviews, so the *Open* and *Open (Read-Only)* commands are not selectable.

2. In the *Library* list box, choose a library name.

The cells in the selected library appear in the *Cell* list box.

To deselect a highlighted library, click the library name.

3. Select *Show Categories*.

The *Category* list box appears between the *Library* and *Cell* list boxes.

The *Category* list box displays the default categories (*Everything* and *Uncategorized*) plus any categories you have added to this library. By default, the *Everything* category is selected.

4. Highlight the category name.

The category name is highlighted. The cells included in that category appear in the *Cell* list box. Hidden cells do not appear in the list.

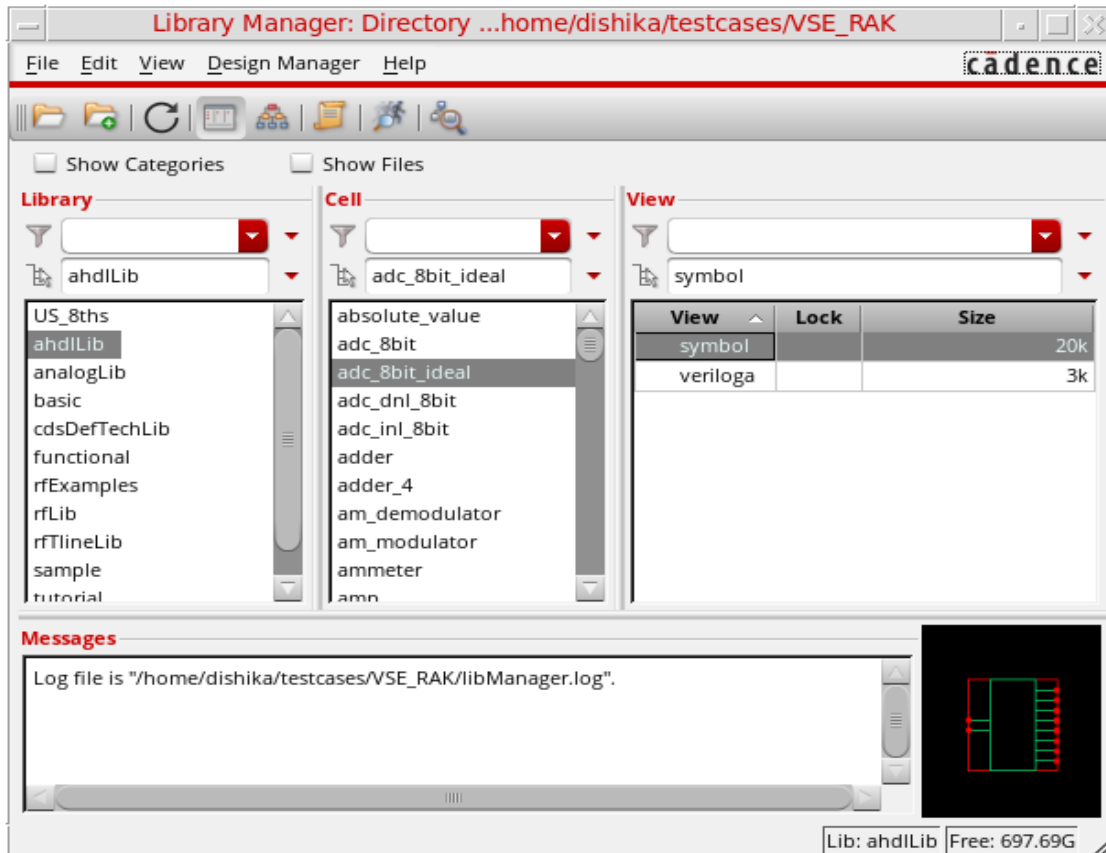
To deselect a highlighted category, click the category name.

5. Choose a cell name.

## Cadence Library Manager User Guide

### Library Management

The cell name is highlighted. The views associated with the cell appear in the *View* list box.



**Note:** To deselect a highlighted cell, click the cell name.

6. Choose the view name of the cellview you want to open.

The view name is highlighted.

**Note:** To deselect a highlighted view, click the view name.

7. Choose *File – Open* or *File – Open (Read-Only)*.

The Library Manager opens the selected cellview.

**Note:** When launched from the Virtuoso environment, the *File* menu shows a list of recently opened cellviews. When launched in standalone mode, it does not show the recently opened cellviews.

### Related Topics

#### Hiding Cells and Showing Hidden Cells

## Creating a New Cellview

To create a new cellview:

1. Open the Library Manager.

**Note:** You can also create a new cellview from the CIW by following the same instructions.

2. Choose *File – New – Cell View*. Alternatively, you can click inside the *Cell* or *View* list box and press `Ctrl+N` on the keyboard.

The New File form is displayed.

You can also type the name of the cell in the *Cell* field and press `Ctrl+N` to open the New File form. In this case, the *Cell* field in the New File form is automatically populated with the name that you have entered in the *Cell* field of the Library Manager form.

3. In the *Library* drop-down list, choose the name of the library in which you want to create a new cellview.
4. In the *Cell* field, type a cell name for the new cellview.

You can set the maximum cell name length allowed by using the `CDS_MAX_CELL_NAME_LENGTH` environment variable.

5. In the *View* field, type a view name for the new cellview.

6. In the *Type* drop-down list, choose the type of view to be opened.
7. In the *Application* section, select the application that should be invoked to display this type of selected cellview.
8. Click *OK*.

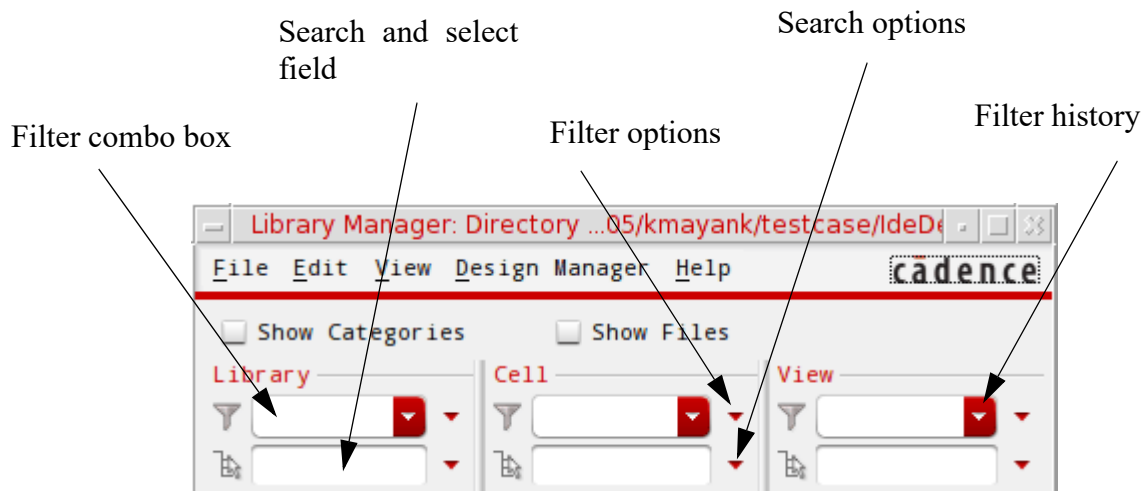
The new cellview appears in a window of the specified product.

### ***Related Topics***

[New File Form](#)

## Filter and Search Options

In the Library Manager window, the *Filter* combo box and the *Select and search* field are provided at the top of the *Library*, *Cell*, and *View* list boxes. Use the *Filter* combo box to specify a filter pattern and accordingly display only a subset of libraries, cells, or views. The *Search and select* field lets you search and select a particular library, cell, or view by specifying a search string.



### Using the Filter Combo Box

The Filter combo box supports use of patterns containing wildcard characters or regular expressions in the specified input. Use of wildcard characters is the preferred and recommended mode of specifying filter patterns. It provides to you advantages that are similar to filtering a list of files from a directory. For example, if you specify *\*Lib\**, the corresponding items, such as *Lib1*, *Lib2*, and *Mylib*, gets displayed.

You can use the `lmgrSetLCVFilter` SKILL function to filter the library, cell, and view list.

To apply filters, specify a filter pattern in the *Filter* combo box and press the *Enter* or *Tab* key. However, if the *Wildcard* option is selected, there is no need to press the *Enter* or *Tab* key. In this case, when you pause, the filter pattern is applied and the *Library*, *Cell*, or *View* list box is updated dynamically.

**Note:** If the *Regular Expression* option is selected, you need to press the *Enter* or *Tab* key to apply filters.

## Cadence Library Manager User Guide

### Library Management

---

Pressing the *Enter* or *Tab* key applies the filter pattern and saves it to the *Filter history* list. If you press the *Tab* key, the cursor moves to the *Search and select* field below the *Filter* combo box.

You can set the filter criteria by selecting the required options from the filter options menu, as shown below.

Field	Description
<b>Filter With</b>	Lets you filter the instances based on one or more specified patterns.
<i>All Of The Words</i>	Sets the filters to display the items that contain all the specified words. For example, if you specify <code>Ana* *cell</code> (separated by space), the corresponding item such as <code>Analog_res_cell</code> gets displayed in the list box.
<i>Any Of The Words</i>	Sets the filters to display all items that have at least one of the specified words. For example, if you specify the filter pattern <code>*mos*</code> , all the corresponding items, such as <code>nmos</code> , <code>pmos_hiVT</code> , and <code>DMOS</code> , is displayed in the list box. This is the default option.
<i>None Of The Words</i>	Sets the filter by specifying the filter pattern that you want to exclude from the list box. For example, if you specify <code>mos</code> , it excludes all the items, such as <code>mos</code> , <code>Mos</code> , and <code>MOS</code> ; whereas, <code>pmos</code> does not get excluded.
<b>Using Case</b>	Lets you determine whether the filter results must be case sensitive, <i>Sensitive Match</i> , or any text case is acceptable, <i>Insensitive Match</i> .
<i>Insensitive Match</i>	Sets the filter to display a subset of items, irrespective of casing. For example, if you specify the text string <code>*mos</code> , all the corresponding items, such as <code>Nmos</code> and <code>CMOS</code> , gets displayed in the list box.
<i>Sensitive Match</i>	Sets the filter to display a subset of items with the exact casing. For example, if you specify the text string <code>*mos</code> , only the lowercase items such as, <code>Nmos</code> and <code>Cmos</code> , gets displayed in the list box. This is the default option.
<b>Match Syntax</b>	The options under <i>Match Syntax</i> let you choose the syntax that is used to interpret a pattern filter.

## Cadence Library Manager User Guide

### Library Management

---

Field	Description
<i>Wildcard</i>	<p>Sets the filter by specifying an item using the following wildcard characters:</p> <ul style="list-style-type: none"><li>❑ <code>*</code> : Matches zero or more characters. For example, if you specify <code>mos*</code>, all the corresponding items starting with <code>mos</code> such as, <code>mos_t</code> and <code>Mos_hiVT</code>, gets displayed in the list box.</li><li>❑ <code>?</code> : Matches any single character. For example, if you specify <code>mos?</code>, only the items followed by a single character such as, <code>mos1</code> and <code>Mos2</code>, gets displayed.</li><li>❑ <code>[]</code> : Represents a set of characters enclosed within the square brackets. For example, if you want to filter the items starting with <code>a</code> or <code>A</code> and containing the substring <code>nalog</code>, specify <code>[aA]nalog</code> as a search string. It displays the matching items, such as <code>analog</code> and <code>Analog</code>.</li><li>❑ <code>\</code> : Escapes any wildcard character. For example, if you want to filter out the results containing “*” (asterisk), you need to add “\” (backslash) before the “*” symbol. For example, if you specify <code>my\*cell*</code>, all corresponding items, such as <code>my*cell</code> and <code>my*cellA</code>, gets displayed in the list box.</li></ul>

This is the default option.



## Cadence Library Manager User Guide

### Library Management

---

Field	Description
<i>Regular Expressions</i>	<p>Sets the filter to use regular expressions to interpret the meaning of a pattern.</p> <ul style="list-style-type: none"> <li>❑ <code>.</code>: If you specify <code>sa.ple</code>, it matches a single character and the corresponding item such as <code>sample</code> gets displayed in the list box.</li> <li>❑ <code>.*</code>: If you specify <code>mos.*</code>, all the corresponding items, such as <code>mos_t</code>, <code>mosA</code>, and <code>Mos_hiVT</code>, gets displayed in the list box.</li> <li>❑ <code>\</code>: If you want to filter out the results containing “*” (asterisk), add “\” (backslash) before the “*” symbol. For example, if you specify <code>my\*cell*</code>, all corresponding items, such as <code>my*cell</code> and <code>my*cellA</code>, gets displayed in the list box.</li> <li>❑ <code>^</code>: If you want to filter the items starting with <code>cds</code>, specify <code>^cds</code>. It displays all the items, such as <code>cdsDefTechLib</code>, <code>cdsDefLibTechview</code>, and <code>cdslibtechView</code>.</li> <li>❑ <code>\$</code>: If you want to filter the items ending with <code>Lib</code>, specify <code>.*Lib\$</code>. It displays all the items, such as <code>analogLib</code> and <code>cdsLib</code>, in the list box.</li> <li>❑ <code>[]</code>: If you specify <code>Lib[1-3]</code>, it displays all the corresponding items, such as <code>Lib1</code>, <code>Lib2</code>, and <code>Lib3</code>, in the list box.</li> <li>❑ <code> </code>: If you want to filter all the items starting with <code>Lib</code> or <code>cds</code>, specify <code>Lib.* cds.*</code>. It displays all the items, such as <code>Lib1</code>, <code>Lib2</code>, and <code>cdsLib</code>.</li> </ul> <p><code>[^]</code> (caret in square brackets): If you want to filter all the items starting with <code>Lib</code> and want to exclude the items containing <code>3</code>, specify <code>^Lib[^3]*</code>. It displays all the items, such as <code>Lib1</code>, <code>Lib2</code>, and <code>Libmgr</code>, except <code>Lib3</code>.</p>

## Cadence Library Manager User Guide

### Library Management

Field	Description
	<p>Some important points to remember:</p> <ul style="list-style-type: none"><li>■ To record the history of previously applied filter patterns, press the <i>Enter</i> or <i>Tab</i> key after specifying the text in the <i>Filter</i> combo box. The history of previously filtered items is saved at the following location:  <code>\$CWD/.cadence/&lt;userID&gt;/libManager</code></li><li>■ To view the history of previously applied filter patterns, click the drop-down button of the <i>Library</i>, <i>Cell</i>, or <i>View</i> combo box. It displays the list of filter pattern history from where you can reapply any of the filters.</li><li>■ The drop-down list maintains a list of the last 50 filtered patterns. The history is valid for all subsequent Virtuoso sessions.</li></ul>
<b>By View Names</b>	This section is included only in the Views filter options menu.
<i>Filter Cells by View Names</i>	Sets the filter to display only those cells that match the filtered views. It is not selected by default.

## Using the Search Option

To search for a specific library, cell, or view, specify a text string in the respective *Search and select* field. To select the instance, press the *Enter* or *Tab* key.

**Note:** If you have already applied a filter, you can only search for the instances from the available subset of library, cell, or view. You can set the search criteria by selecting the required options from the *Search options* menu, as shown below.

Field	Description
<b>Match Name</b>	Allows you to search for an instance either by specifying a substring or a prefix.
<i>Substring</i>	Searches the first instance in the list box that contains the specified text string. For example, if you specify <code>mos</code> , the <code>Nmos</code> instance is selected in the list box.

## Cadence Library Manager User Guide

### Library Management

---

Field	Description
<i>Prefix</i>	<p>Searches the first instance in the list box that starts with the specified text string. For example, if you specify <code>mos</code>, the <code>mosView</code> instance is selected in the list box.</p> <p>This is the default option.</p>
<b><i>Using Case</i></b>	<p>The options under <i>Using Case</i> let you determine whether search results must be case sensitive (<i>Sensitive Match</i>) or any text case is acceptable (<i>Insensitive Match</i>).</p>
<i>Insensitive Match</i>	<p>Searches the first instance in the list box irrespective of casing. For example, if you specify the text string <code>mos</code>, the <code>Mos</code> instance gets selected in the list box.</p>
<i>Sensitive Match</i>	<p>Searches the first instance with the exact casing. For example, if you specify the text string <code>mos</code>, only the lowercase items, such as <code>Nmos</code> or <code>Cmos</code>, gets selected in the list box and items like <code>CMOS</code> does not get matched.</p> <p>This is the default option.</p> <p>Once you set the filter in the <i>List</i> mode, the <i>Tree</i> mode only displays the items matching the filter criteria.</p> <p>To view the results in the <i>Tree</i> mode, select <i>View – Tree</i>. In this mode:</p> <ul style="list-style-type: none"><li>■ When you choose a library, only the libraries matching your filter string appear in the expanded branch.</li><li>■ When you choose a cell, only the cells matching your filter string appear in the expanded branch.</li><li>■ When you choose a view, only the views matching your filter string appear in the expanded branch.</li></ul>

### ***Related Topics***

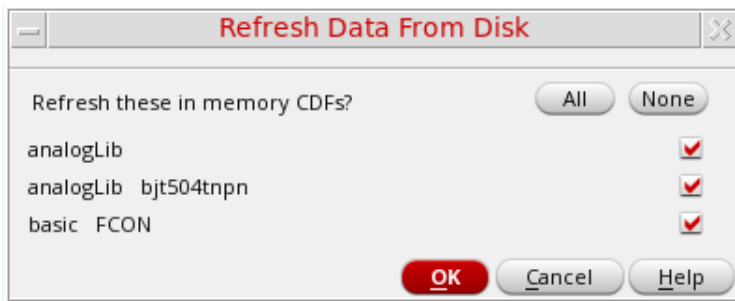
[Library Manager Form](#)

[ImgrSetLCVFilter](#)

## Refreshing the View and Data

To refresh the view:

1. Choose *View – Refresh* in the Library Manager window.



By default, all modified libraries appear selected. If a cellview is opened in edit mode, the corresponding entry does not get listed.

2. Deselect any libraries that you do not want to update in the view and click *OK*.

The selected libraries are reloaded from the library definitions.

To refresh design data, technology files, and CDF data in the current session:

- Select *File – Refresh* in the Virtuoso Command Interpreter Window (CIW).

If there is no new data to refresh, an appropriate message appears.

### ***Related Topics***



[Library Manager Form](#)

[Viewing the cds.lib Updates](#)

## Viewing the Current Cellview Status

To view the current cellview status:

- Click the *Show open cellviews in use*  button on the Library Manager toolbar.

The lists in the window are reloaded with icons next to names. The icons indicate whether a library, cell, or view is opened in read mode (green icon ) or edit mode (red icon ) .

You can click the toggle button *Show open cellviews in use* to hide the status icon indicators.

### ***Related Topics***

[Library Manager Form](#)

## Renaming a Library

To rename a library:

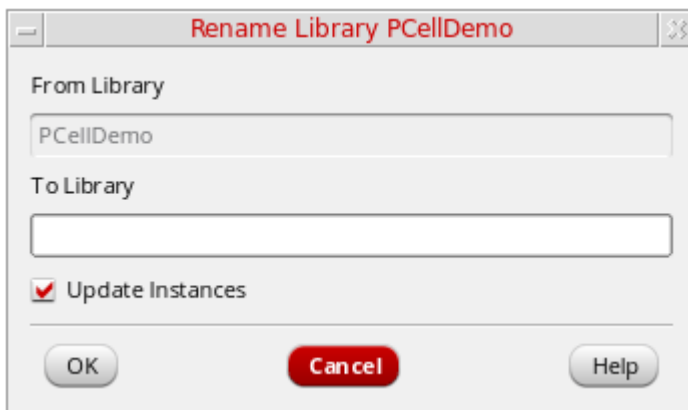
1. Open the Library Manager.
2. Select the library you want to rename.

Make sure no cell name or view name is selected. To deselect all cells and views, right-click in the *Library* list box.

To set preferences for *Edit – Copy* and *Edit – Rename*, see [Setting Copy and Rename Preferences](#).

3. Choose *Edit – Rename*.

The Rename Library form appears.



The screenshot shows a dialog box titled "Rename Library PCellDemo". It has two text input fields: "From Library" which contains the text "PCellDemo", and "To Library" which is currently empty. Below these fields is a checkbox labeled "Update Instances" which is checked. At the bottom of the dialog are three buttons: "OK", "Cancel", and "Help".

The name of the selected library appears in the *From Library* field. The *Update Instances* check box is selected by default.

- ❑ When *Update Instances* is selected, the Library Manager replaces all instances by the name in the *From Library* field with the name in the *To Library* field.

For example, with *Update Instances* selected, an instance of `.../projectLib/iopin/symbol` is renamed to `.../myLib/iopin/symbol`. (All instances of `projectLib` are changed to `myLib`.)

- ❑ When *Update Instances* is unselected, the software leaves references to the *From Library* name unchanged.

If you rename `projectLib` to `myLib` with *Update Instances* unselected, instances of `.../projectLib/iopin/symbol` (for example) remain the same such that they become unbound instances unless you replace the `projectLib` library.

## Cadence Library Manager User Guide

### Library Management

---

4. In the *To Library* field, type a new name for the library.

**Note:** The library name cannot duplicate another library name. If the name you type in the *To Library* field already exists (such as `analogLib`), an error message appears.

5. Click *OK*.

The system changes the name of the selected library (*From Library*) to the new name (*To Library*).

**Note:** If the destination library already contains the `tech.db` file, the source library's `tech.db` is not copied. Otherwise, while copying a source library to an existing library, the `tech.db` file associated with the source library also get copied to the destination library.

## Renaming a Cell

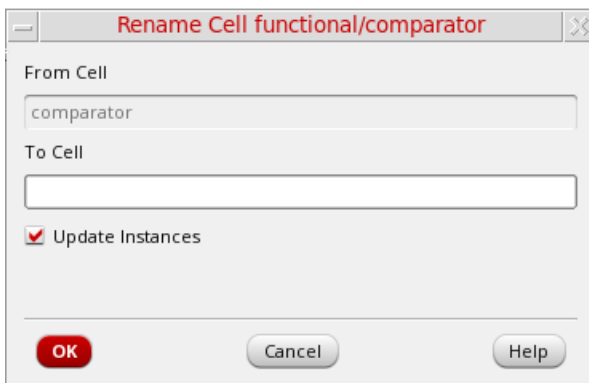
To rename a cell:

1. Select the cell you want to rename.

Make sure no view name is selected. (If you are in *View – Lists* mode, you can deselect all views by right-clicking in the *Cell* list box.)

2. Choose *Edit – Rename*.

The Rename Cell form appears.



The screenshot shows a dialog box titled "Rename Cell functional/comparator". It has two text input fields: "From Cell" which contains the text "comparator", and "To Cell" which is currently empty. Below these fields is a checkbox labeled "Update Instances" which is checked. At the bottom of the dialog are three buttons: "OK" (highlighted in red), "Cancel", and "Help".

The name of the selected cell appears in the *From Cell* field.

**Note:** If the selected cell is a combined cell, the *Edit – Rename* command is not available. See [Creating and Displaying a Combined Library](#) for more information.

3. In the *To Cell* field, type the new name for the cell.

The cell name cannot duplicate an existing cell name in the library.

You can set the maximum cell name length allowed by using the `CDS_MAX_CELL_NAME_LENGTH` environment variable.

4. Deselect *Update Instances*.

- ☐ When *Update Instances* is selected (the default), the Library Manager replaces all instances having the *From Cell* name with the *To Cell* name.
- ☐ When *Update Instances* is unselected, the Library Manager keeps references to the *From Cell* name such that all placed instances of the *From Cell* continue to reference that name: Instances using the old (*From Cell*) name are unbound instances unless you replace them with the new (*To Cell*) cell.

5. Click *OK*.



## Cadence Library Manager User Guide

### Library Management

---

The Library Manager changes the name of the selected cell (*From Cell*) to the new name (*To Cell*).

#### ***Related Topics***

[Opening the Library Manager](#)

[Rename Library Form](#)

[Rename View Form](#)

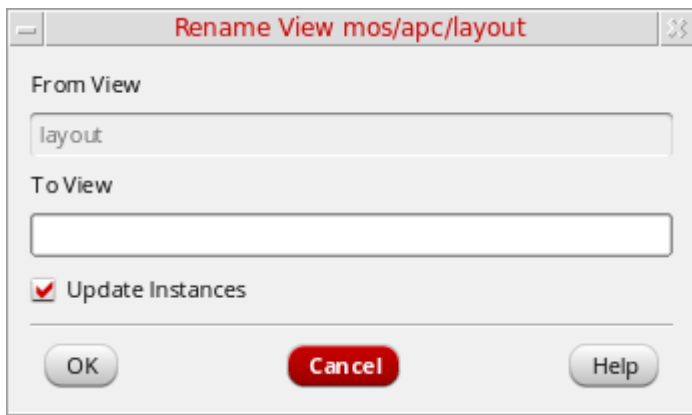
[Renaming a Reference Library](#)

## Renaming a View

To rename a view:

1. Select the view you want to rename.
2. Choose *Edit – Rename*.

The Rename View form appears.



The screenshot shows a dialog box titled "Rename View mos/apc/layout". Inside, there are two text input fields: "From View" which contains the text "layout", and "To View" which is currently empty. Below these fields is a checkbox labeled "Update Instances" which is checked. At the bottom of the dialog are three buttons: "OK", "Cancel", and "Help".

The name of the view you selected appears in the *From View* field. The entire cellview path appears in the title banner of the form.

3. In the *To View* field, type the new name for the view.

The new view name cannot duplicate an existing view name for the cell.

4. Deselect *Update Instances*.

- ☐ When *Update Instances* is selected, the Library Manager overwrites any occurrences of the *From View* name with the *To View* name.

If you rename `symbol` to `symbolA` (for example), all instances of `.../projectLib/buff/symbol` are changed to `.../projectLib/buff/symbolA`.

- ☐ When *Update Instances* is unselected, the Library Manager keeps references to the *From View* name.

If you rename `symbol` to `symbolA` with *Update Instances* unselected, instances of `.../projectLib/buff/symbol` (for example) remain the same such that they become unbound instances unless you replace the `symbol` view.

5. Click *OK*.

## Cadence Library Manager User Guide

### Library Management

---

The Library Manager changes the name of the selected view (*From View*) to the new name (*To View*).

If you are working with a design-managed library, the Rename Library dialog box shows an additional section.

#### ***Related Topics***

[Opening the Library Manager](#)

[Rename Library Form](#)

[Rename Cell Form](#)

[Rename View Form](#)

[Renaming a Reference Library](#)

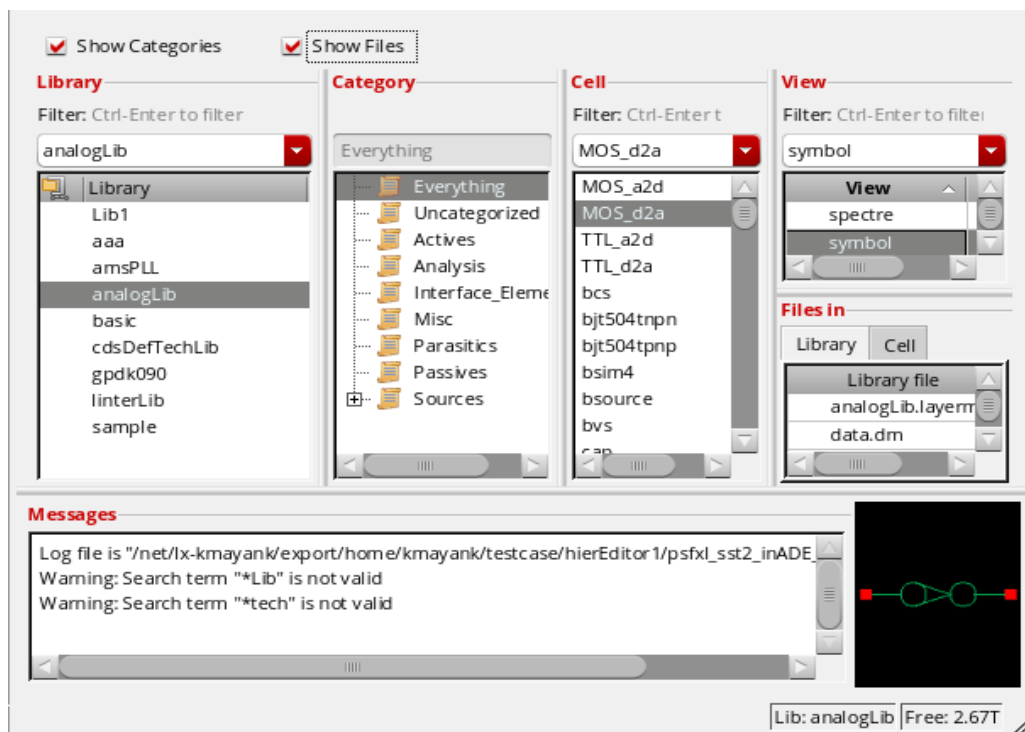
## Renaming Files

To rename a file:

1. Depending on your viewing mode:

- ☐ If you are in *View – Lists* mode, select the *Show Files* check box.

The *Files In Library* and *Files In Cell* list boxes appear on the Library Manager form.



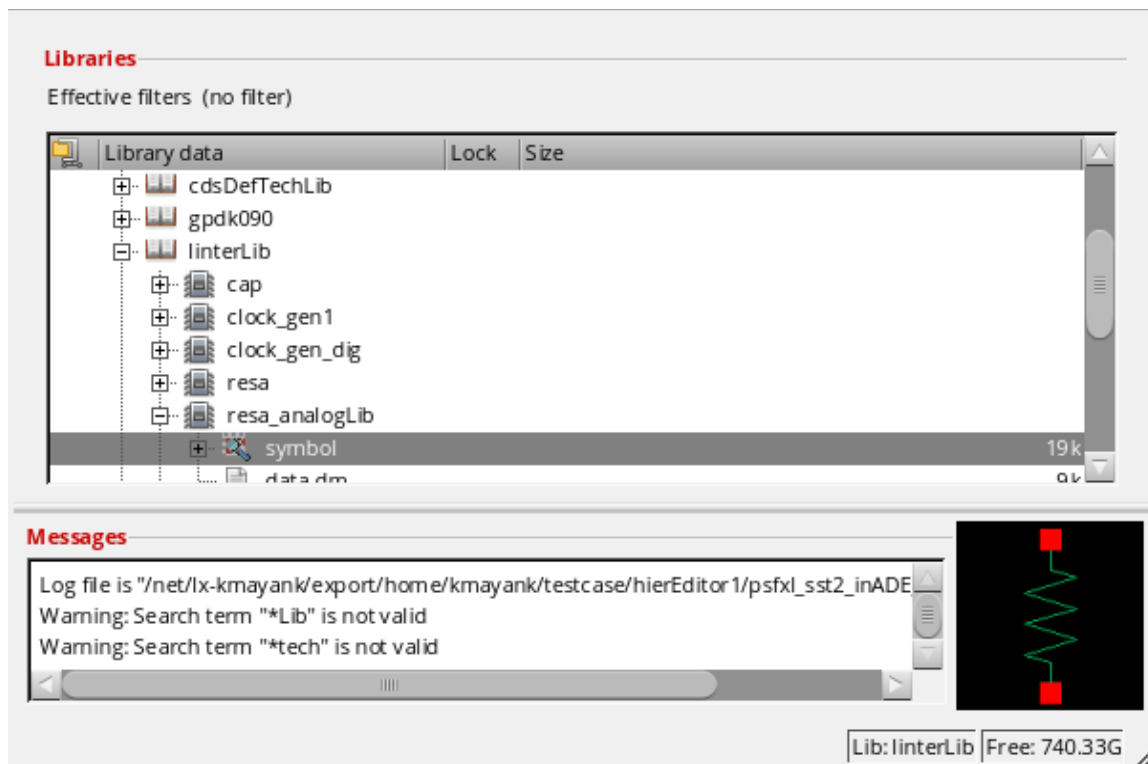
- ☐ If you are in *View – Tree* mode, expand the branches of the tree until you can see the file you want to rename.

2. Select the file you want to rename.

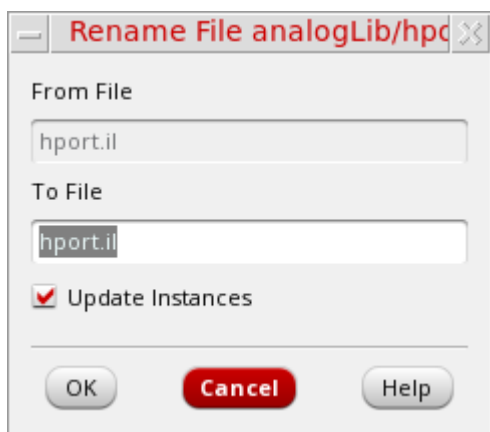
3. Choose *Edit – Rename*.

## Cadence Library Manager User Guide

### Library Management



The Rename File form appears. The name of the file you selected appears in the *From File* field.



4. In the *To File* field, type the new name for the file.

The new file name cannot duplicate an existing file name in the library.

5. Click *OK*.

## Cadence Library Manager User Guide

### Library Management

---

The Library Manager changes the name of the selected file (*From File*) to the new name (*To File*).

If you are working with a design-managed file, the Rename Cell File dialog box shows an additional section.

#### ***Related Topics***

[Controlling the Display of Library Information](#)

[Rename File Form](#)

[Renaming a Reference Library](#)

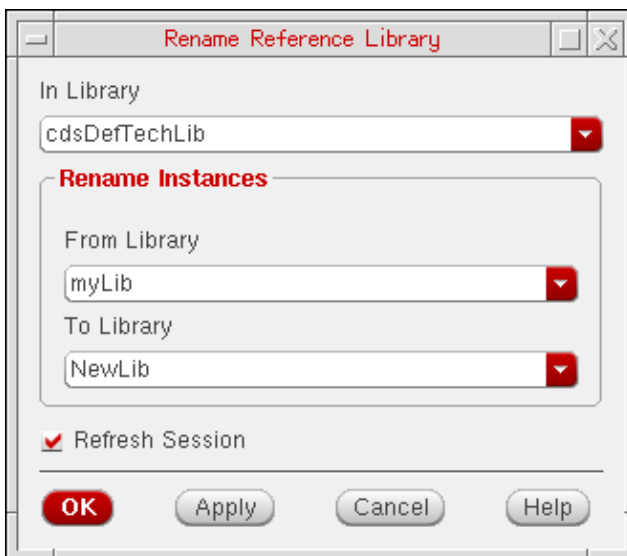
## Renaming a Reference Library

You can change the reference library for instances in your design library. For example, if your design library, `newLib`, contains `via` cells from the reference library `basic`, you can change the reference library from the `basic` library to the `analogLib` library so that the `via` cells, and any other cells in your design library that reference the `basic` library, now reference the `analogLib` library. This assumes that the `analogLib` library contains a `via` cell.

To change a reference library for a design library:

1. Select the library that contains references you want to rename.
2. Choose *Edit – Rename Reference Library*.

The Rename Reference Library form appears.



3. In the *In Library* drop-down list, select or type the name of the design library that uses a reference library whose name you want to change. In the *From Library* drop-down list, select or type the name of the current reference library.
4. In the *To Library* drop-down list, select or type the name of the new reference library.
5. Click *OK* to save the changes and close the Rename Reference Library form.

**Note:** To apply changes and keep the form open, click *Apply*.

## Cadence Library Manager User Guide

### Library Management

---

The system changes the name of the current reference library to the name of the new reference library for all instances in the specified design library.

If you cannot rename the reference library, it might be for one of the following reasons:

- An input parameter is invalid.
- The system cannot automatically open the design library for writing.
- You cannot get access permission to the design library.

#### ***Related Topics***

[Rename Reference Library Form](#)

[Viewing and Changing File Permissions](#)



## Changing Library Reference

There might be instances where you need to map all the instance from multiple libraries to a specified destination library. In such cases, you can use the *Change Library References* feature that enables you to map multiple libraries to the specified destination library at once.

For example, in design library, `test`, you can change the references of libraries, `cdsDefTechLib` and `pc` to the `pc` destination library, as shown below:



## Cadence Library Manager User Guide

### Library Management

To change a reference library for a design library:

1. Select the library that contains references you want to change.
2. Choose *Edit – Change Library Reference*.

The Change Library References form appears.

**Change Library References**

**Change References**

**In Library**

PDK\_devices  
analogLib  
basic  
cdn20  
cdn20\_abutment  
cdn20\_testing

Filter: \*

**From Library**

<All>  
PDK\_devices  
analogLib  
basic

Filter: \*

**To Library**

<As Is>  
PDK\_devices  
analogLib  
basic

Filter: \*

**From View:** <All> **To View:** <As Is>

☐ Show Selected Libraries Only **Add** **Replace**

**Change List**

From Library	From View	To Library	To View
--------------	-----------	------------	---------

**0 reference changes**

☐ Include unbound referencing  
Include unbound instances in a design when changing references.  
Use this with caution as it might also result in instances becoming unbound after the change.

**OK** **Cancel** **Apply** **Help**

3. In the *In Library* list box, select the name of the design library that uses a reference library whose name you want to change.
4. In the *From Library* list box, select the name(s) of the current design libraries.
5. In the *From Library* list box, select the name of the new reference library.

6. Click *Add*. The change list gets added in the *Change List* area.

7. Click *OK*.

The system changes the references of the current libraries to the name of the new reference library for all instances in the specified design library.

### ***Related Topics***

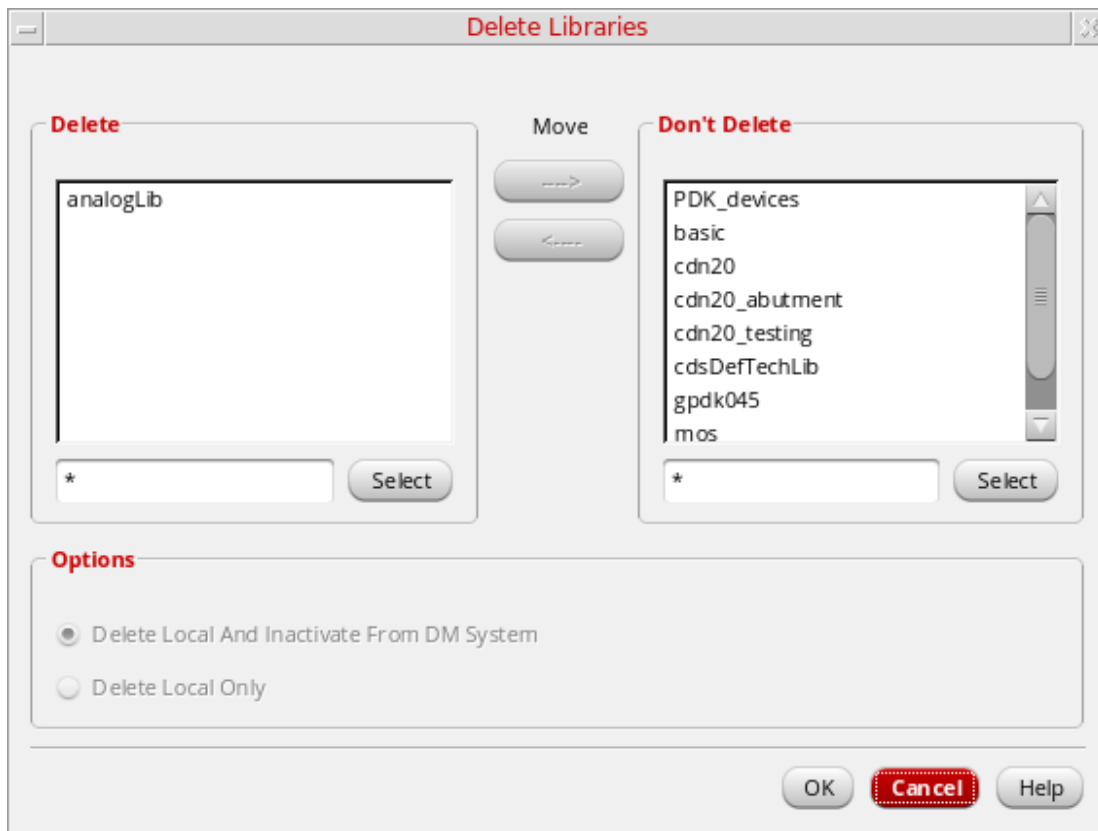
[Rename View Form](#)

## Deleting a Library

To delete a library from the Library Manager:

1. Select the library you want to delete.
2. Choose *Edit – Delete*.

The Delete Libraries form appears. The selected library appears in the *Delete* list box.



3. To delete additional libraries:
  - a. Select a library from the *Don't Delete* list box.
  - b. Click the left-facing *Move* arrow to move the selected libraries to the *Delete* list box.



### Tip

You can double-click a single item in the *Don't Delete* list box to move it over to the *Delete* list box. You can select multiple items by *Shift*-clicking, *Ctrl*-clicking, clicking and dragging, or using the *Select* filter.

4. Select one of the following *Options*:

- ☐ *Delete Local And Inactivate From DM System*: Deletes a local copy of a library and the copy in the design management repository.
- ☐ *Delete Local Only*: Deletes only your local copy of a library (not the checked in copy in the design management repository).

5. Click *OK* and then *Yes* in the confirmation dialog to perform the delete operation.

### ***Related Topics***

[Delete Libraries Form](#)

[Deleting a Cell](#)

[Deleting a View](#)

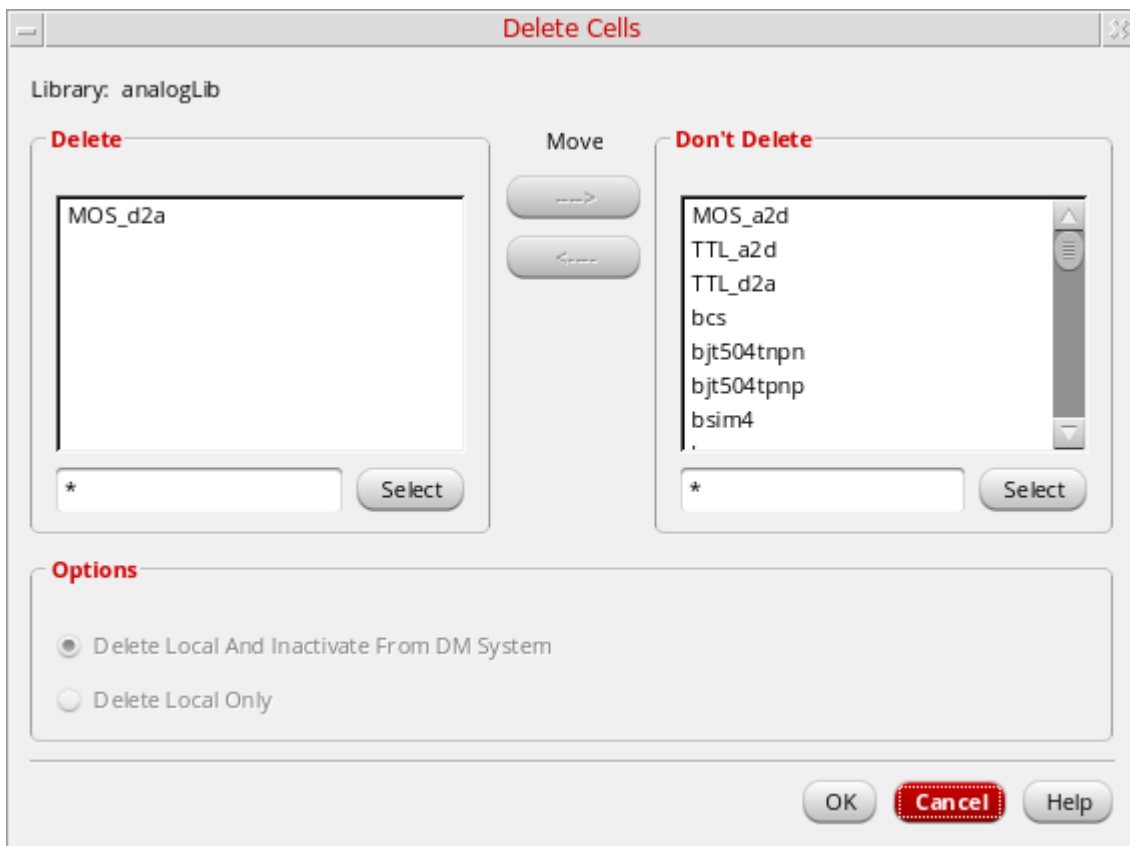
## Deleting a Cell

To delete a cell from the Library Manager:

1. Select the cell you want to delete.
2. Choose *Edit – Delete*.

The Delete Cells form appears.

The selected cell appears in the *Delete* list box.



**Note:** If the cell is selected from a combined library, then it does not appear by default in the *Delete* list. The cell appears in the *Don't Delete* list and the cell name indicates the physical library that it belongs to. If you want to delete the cell, move it to the *Delete* list. Also, the cell gets deleted from the physical library to which it belongs.

**Note:** If the cell appears in multiple libraries under the combined library, the *Don't Delete* list displays an entry for each library, the library name is displayed with the cell name to identify the library, so that you can select the ones you want to delete. However,

if the cell is physically in the top-level combined library, then it does appear in the *Delete* list and all the other cells that are in that library are displayed in the *Don't Delete* list.

3. To delete additional cells, you need to select a cell from the *Don't Delete* list box and click the left-facing *Move* arrow to move the selected cells to the *Delete* list box.
4. Select one of the following *Options*:
  - ☐ *Delete Local And Inactivate From DM System*: Deletes a local copy of a cell and the copy in the design management repository.
  - ☐ *Delete Local Only*: Deletes only your local copy of a cell (not the checked in copy in the design management repository).
5. Click *OK* and then click *Yes* to perform the specified delete operation.

### ***Related Topics***

[Creating and Displaying a Combined Library](#)

[Delete Cells Form](#)

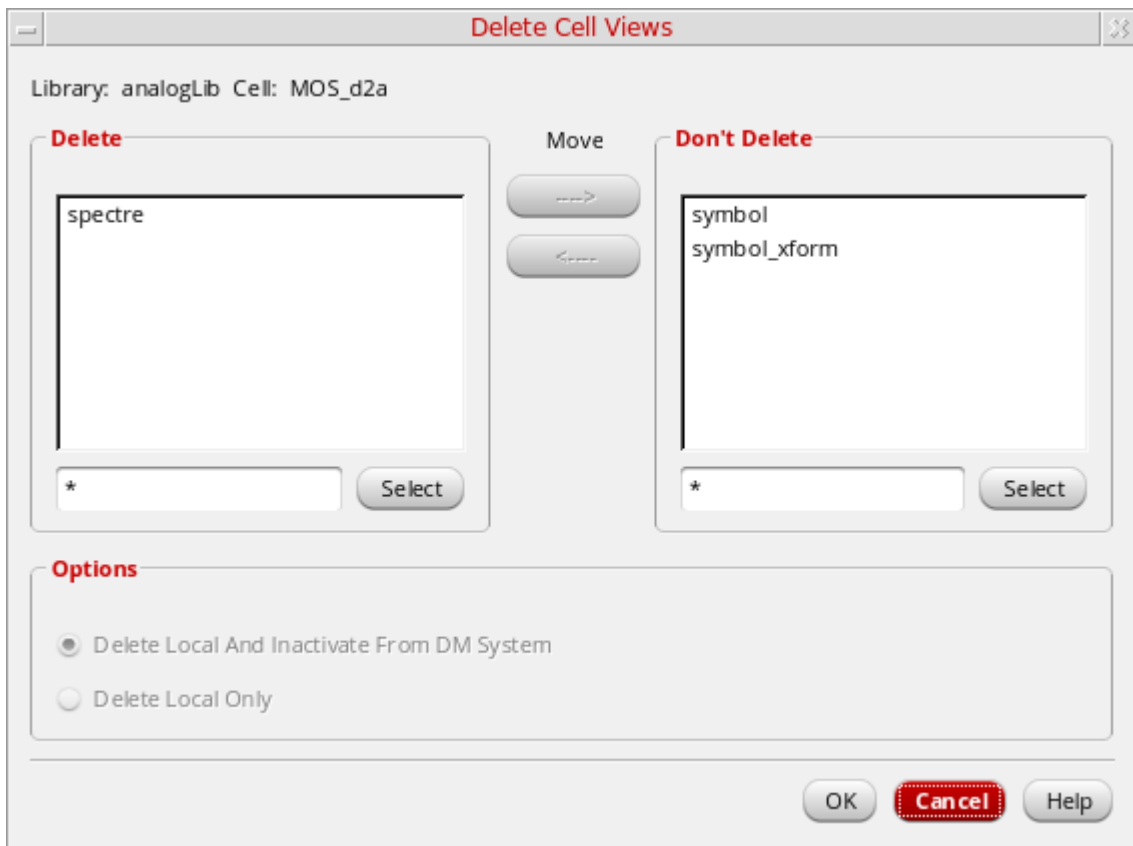
## Deleting a View

To delete a view from the Library Manager:

1. Select the view you want to delete.
2. Choose *Edit – Delete*.

The Delete Cell Views form appears.

The selected view appears in the *Delete* list box.



**Note:** If the view is selected from a combined library, then it does not appear by default in the *Delete* list. It appears in the *Don't Delete* list and its name indicates the physical library that it belongs to. If you want to delete the view, move it to the *Delete* list. In addition, the view gets deleted from the physical library to which it belongs.

**Note:** If the view is part of a combined cell, that is, a cell that is found in more than one library in the combined library, the *Don't Delete* list displays an entry for each library, so that you can select the ones you want to delete. However, if the cell is a part of the



top-level combined library, then the view does appear in the *Delete* list and all the other views of that cell are displayed in the *Don't Delete* list.

3. To delete additional views, you need to select a view from the *Don't Delete* list box and click the left-facing *Move* arrow to move the selected views to the *Delete* list box.
4. Select one of the following *Options*:
  - ☐ *Delete Local And Inactivate From DM System*: Deletes a local copy of a view and the copy in the design management repository.
  - ☐ *Delete Local Only*: Deletes only your local copy of a view (not the checked in copy in the design management repository).
5. Click *OK* and then *Yes* in the confirmation dialog to perform the delete operation.

### ***Related Topics***

[Delete Cell Views Form](#)

[Deleting Cells Using Filters](#)

[Deleting Library or Cell Files](#)

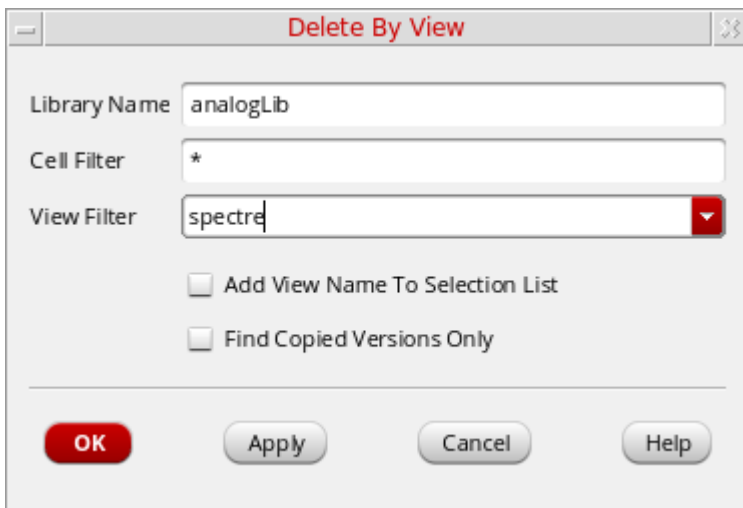
## Deleting Cells Using Filters

You can delete cellviews or group of cells from your local directory or from both the local directory and the current design management repository. You can delete all views for a cell or only those views for versions you have copied.

To delete views for a given cell:

1. Select the view you want to delete.
2. Choose *Edit – Delete By View*.

The Delete By View form appears . The name of the selected library appears in the *Library Name* field.



3. In the *Cell Filter* field, type a filter string for the cells you want to delete.

For example, type \* to indicate all cells or p\* to indicate all cells beginning with a lowercase p.

4. In the *View Filter* drop-down list, type a filter string for the views of these cells you want to delete, or select a view name from the drop-down list.



### Tip

To add a new view name to the View Filter drop-down list selection list:

- a. Select the *Add View Name To Selection List* check box.
- b. Click *Apply*.

## Cadence Library Manager User Guide

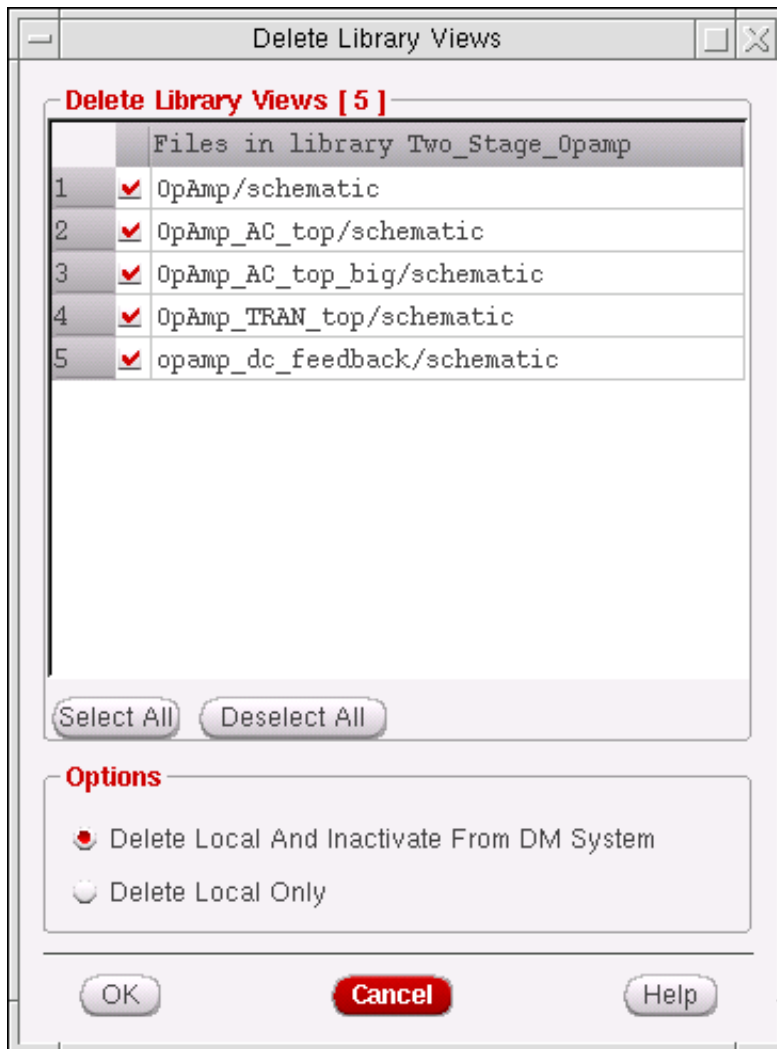
### Library Management

---

The new name is added to the bottom of the *View Filter* drop-down list.

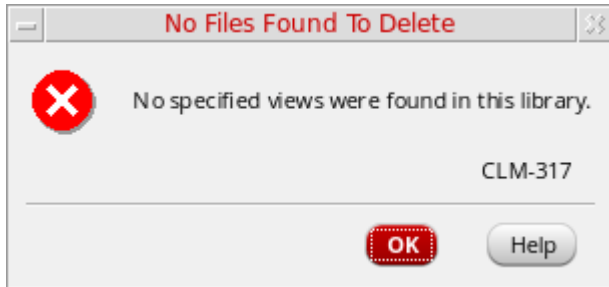
5. If you want to delete only cells and views that you copied previously, select the *Find Copied Versions Only* check box.
6. Click *OK*.

The Delete Library Views form appears. The cellviews that match the criteria you specified on the Delete By View form.



**Note:** If no cellviews that match the criteria are found, the No Files Found To Delete form appears. Click *OK* to close this form and return to the Delete By View form to specify new

criteria.



7. On the Delete Library Views form, select one of the following *Options*:

- ☐ *Delete Local Only*: Deletes the local copy of a cellview only.
- ☐ *Delete Local And Inactivate From DM System*: Deletes the local copy of a cellview and the copy in the current design management repository.

8. Click **OK**.

The selected cellviews are deleted.

The following are some possible circumstances that might generate error messages when you try to delete a cellview from a library:

- If you try to delete a read-only library, an error message appears, indicating that the process of deleting by view failed.
- In rare instances, the MPS (Message Passing Subsystem) server (used by the Virtuoso Studio design environment) that stores the directory for the library you specified might be down. In this case, an error message appears, indicating that the library from which you want to delete a view is not found.
- If you do not specify a cell or view in the *Cell Filter* or *View Filter* fields, an error message appears, indicating that a cell or view name is missing.
- If you specify a name for a library, cell, or view that does not exist, an error message appears, indicating that no files were found.

### ***Related Topics***

[Delete By View Form](#)

[Delete Library Views Form](#)

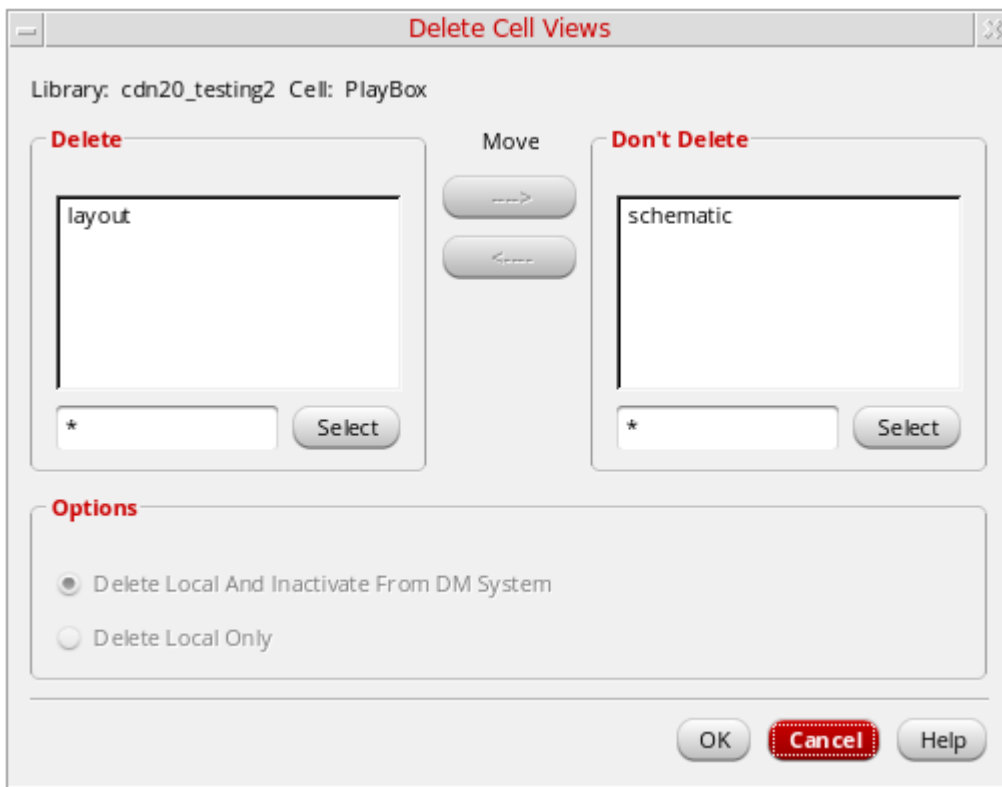
## Deleting Library or Cell Files

To delete library-level or cell-level files from the Library Manager:

1. Select the file you want to delete in the *Files in Library* or *Files in Cell* list box.
2. Choose *Edit – Delete*.

The Delete Library Files or Delete Cell Files form appears.

The selected file appears in the *Delete* list box.



If the file is selected from a combined library, then it does not appear by default in the *Delete* list. It appears in the *Don't Delete* list and its name indicates the physical library that it belongs to.

To delete a file, move it to the *Delete* list, the file gets deleted from the physical library to which it belongs. If the file is found in multiple libraries, the *Don't Delete* list displays an entry for each library, you can select the ones you want to delete. However, if the file is part of the top-level combined library, then it does appear in the *Delete* list and all the other files in the library (or cell) are displayed in the *Don't Delete* list.

For more information about combined libraries, see [Creating and Displaying a Combined Library](#).

**3.** To delete additional files:

- a.** Select a file from the *Don't Delete* list box.
- b.** Click the left-facing *Move* arrow to move the selected files to the *Delete* list box.

You can double-click a single item in the *Don't Delete* list box to move it over to the *Delete* list box. You can select multiple items by *Shift*-clicking, *Ctrl*-clicking, clicking and dragging, or using the *Select* filter.

**4.** Select the desired option from the *Options* field.

**5.** Click *OK* and then click *Yes* to perform the specified delete operation.

***Related Topics***

[Delete Cell Views Form](#)

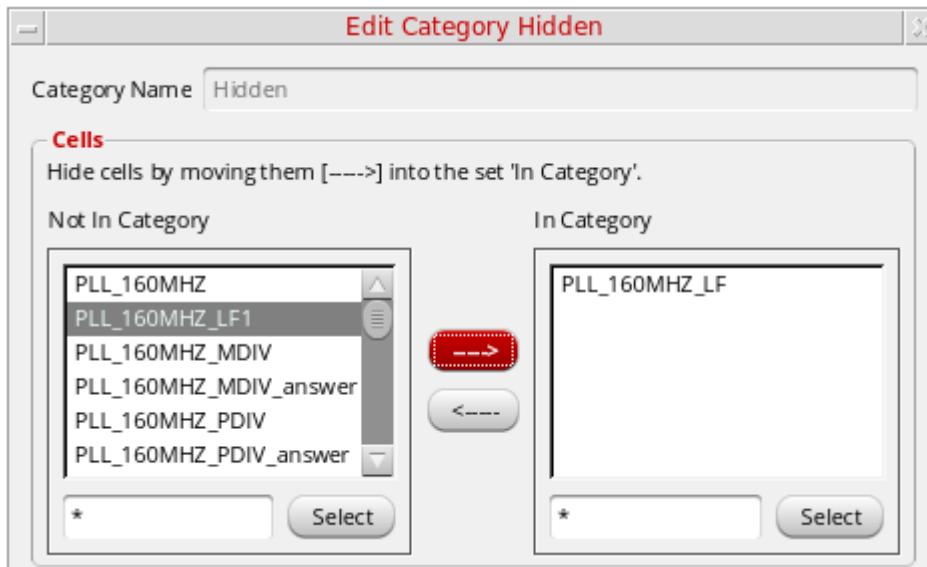
## Hiding Cells and Showing Hidden Cells

You can hide a cell from the *Cell* list box of the Library Manager window. To do this, you need to perform the following steps:

1. Right-click the cell and select the *Hide Cell* option from the context-sensitive menu. Alternatively, you can select *Edit – Hide Cell*.

The *Hide Cell* command is disabled if the library is in read-only mode.

The Edit Category Hidden window is displayed. In the *Cells* section of this window, move the cell from *Not in Category* list box to *In Category* list box using the right arrow button.



2. Click *OK*.

The new *Hidden* category is created in the *Category* list box of the Library Manager window.

Alternatively, use the SKILL function [ddRegHiddenCellsFunc](#) to hide cells.

In case you hide the cells that are already placed in a design, you still can:

- Edit the existing instances
- Descend the existing instances
- View the existing instances in the netlist
- View the existing instances in LVS

## Showing Hidden Cells

If you have the required permissions, you can show the hidden cells by performing the following steps:

1. Right-click the *Hidden* category and select *Modify* from the context-sensitive menu. The Edit Category Hidden window is displayed.
2. In the *Cells* section, move the cell from *In Category* list box to *Not In Category* list box using the left arrow button.
3. Click *OK*.

### ***Related Topics***

[Opening a Cellview](#)



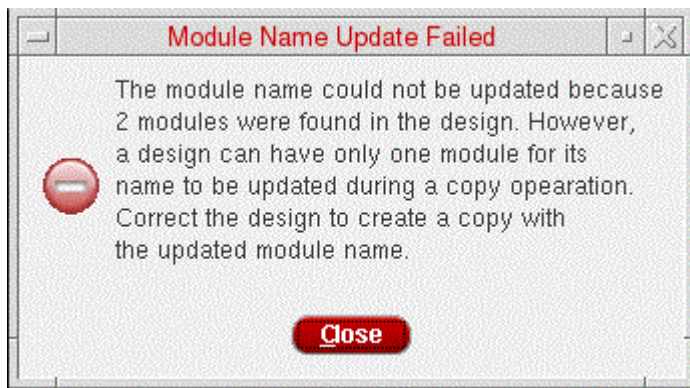
## Text Cellviews

While copying or renaming a text cellview, the corresponding module name in the HDL file can be updated automatically using the `autoModuleNameUpdate` cdsenv variables.

For example, if you rename functional text cellview `myVerilogCell`, with the module by the same name, to `myNewVerilogCell`, the module name updates in the Verilog file automatically.

While copying or renaming a text cellview, the match is done only till the first uncommented module name is found instead of matching till "`;`".

The module name does not get updated automatically if there are HDL file parsing errors or multiple modules are existing in the HDL file of the associated cell. In case the HDL file has multiple modules, copying or renaming the associated cell displays the following error message and the module name does not get updated.



The feature to update the module name in the HDL file automatically is available for Verilog, SystemVerilog, Verilog-A, or Verilog-AMS views.

For Verilog-A views, matching of cell names and module names is not done by default. However, you can reconfigure this behavior by setting the `matchModuleNameCellName` cdsenv variable.

### ***Related Topics***

[Library Manager Environment Variables](#)

[Opening a Cellview](#)

[Creating a New Cellview](#)

## Editing Library Properties

To edit the properties associated with a library:

1. In the CIW, choose *Tools – Library Manager*.

The Library Manager form appears.

2. Select the library whose properties you want to edit.

3. Choose *Edit – Properties*.

The Library Property Editor form appears. The properties of the selected library appear at the bottom of the form.

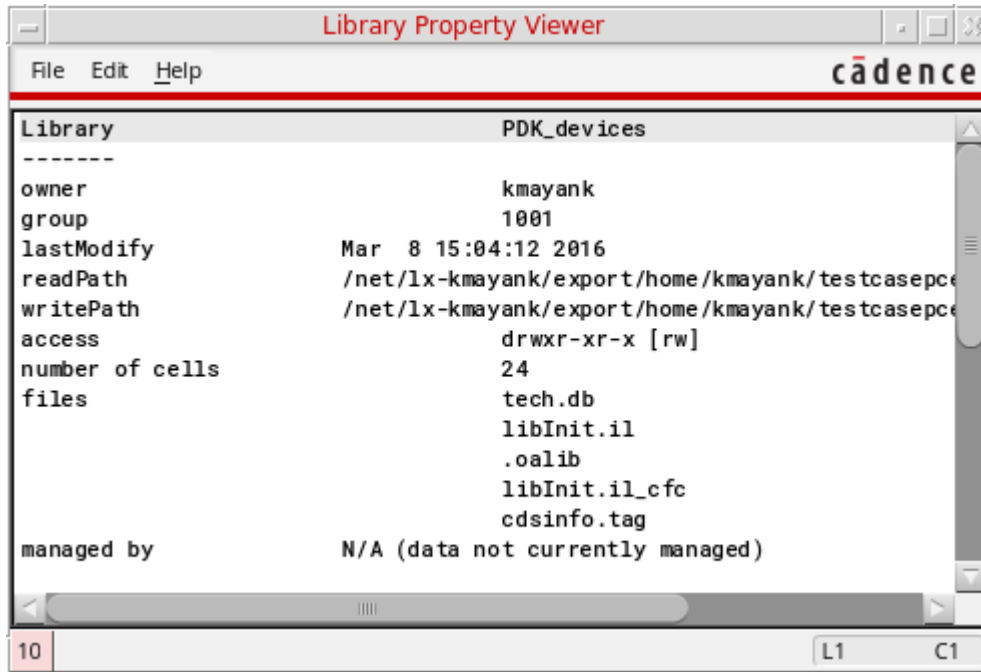
The screenshot shows the 'Library Property Editor' dialog box. It has a title bar with a close button (X). The dialog is divided into three main sections: 'Library', 'UNIX Permission Mode', and 'Properties'.  
The 'Library' section contains fields for Name (tutorial), Owner (jishika), Group (1001), Last Modify (Sep 11 13:23:07), Read Path (scratch02/jishika/testcases/VSE\_RAK/tutorial), and Write Path (scratch02/jishika/testcases/VSE\_RAK/tutorial).  
The 'UNIX Permission Mode' section contains fields for Owner (rwx), Group (r-x), and Other (r-x).  
The 'Properties' section contains a list of properties: attachedFiles (with an EXPAND button), cdfData (with a text field containing 'ematic" instDisplayMode "instName"))'), minGridResolution (with an EXPAND button), nameSpaceUpdated (with a date field showing 'Oct 4 06:36:14 1995'), and viewProps (with an EXPAND button).  
At the bottom of the dialog are buttons for OK, Cancel, Apply, Add, Delete, Modify, and Help.

## Cadence Library Manager User Guide

### Library Management

---

If you do not have write permission for the selected library, you can view but not edit the properties. The Library Property Viewer appears instead of the Library Property Editor.



4. In the Library Property Editor form, you can add, delete, and modify library properties:

- ☐ To add a property, click *Add*.
- ☐ To delete a property, select the property and click *Delete*.  
The selected property is removed from the form.
- ☐ To modify a property, select the property and click *Modify*.

5. Click *Apply*.

### ***Related Topics***

[Library Manager Form](#)

[Library Property Editor Form](#)

[Adding Properties to a Library, Cell, or View](#)

[Modifying Properties of a Library, Cell, or View](#)

## Editing Cell Properties

To edit the properties associated with a cell:

1. On the Library Manager form, select the cell whose properties you want to edit.
2. Choose *Edit – Properties*.

The Cell Property Editor form appears. The properties of the selected cell appear at the bottom of the form.

The screenshot shows the 'Cell Property Editor' dialog box. It has a title bar with a close button (X). The dialog is divided into three main sections: 'Cell', 'UNIX Permission Mode', and 'Properties'. The 'Cell' section contains fields for Name (adder), Owner (empty), Group (1001), Last Modify (Sep 11 13:31:24), Read Path (r02/[redacted]/testcases/VSE\_RAK/tutorial/adder), and Write Path (r02/[redacted]/testcases/VSE\_RAK/tutorial/adder). The 'UNIX Permission Mode' section contains fields for Owner (rwx), Group (r-x), and Other (r-x). The 'Properties' section contains a text area with the value 'cdfData belSet "v i pwr" paramLabelSet "k1 k2 model")'. At the bottom, there are buttons for OK, Cancel, Apply, Add, Delete, Modify, and Help.

If you do not have write permission for the selected cell, you can display but not edit the properties. The Cell Property Viewer appears instead of the Cell Property Editor.

3. In the Cell Property Editor form, you can add, delete, and modify cell properties:

- ☐ To add a property, click *Add*.
- ☐ To delete a property, select the property and click *Delete*.

The selected property is removed from the form.

- ☐ To modify a property, select the property and click *Modify*.

4. Click *Apply*.

***Related Topics***

[Cell Property Editor Form](#)

[Adding Properties to a Library, Cell, or View](#)

[Modifying Properties of a Library, Cell, or View](#)

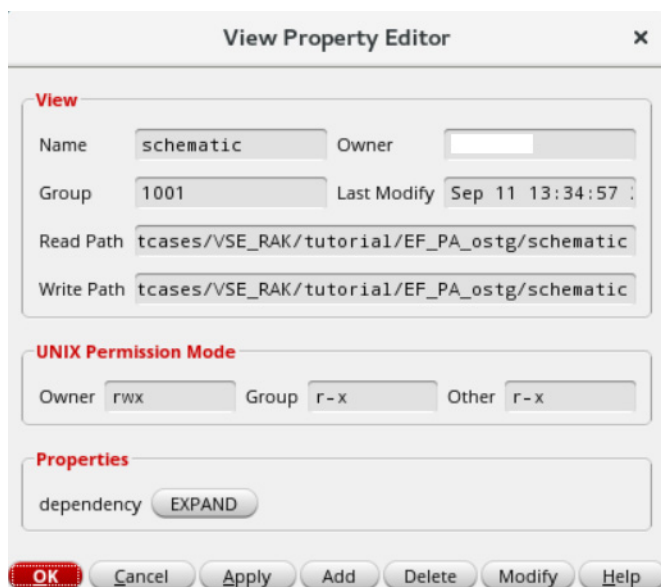
## Editing View Properties

To edit the properties associated with a view:

1. On the Library Manager form, select the view whose properties you want to edit.
2. Choose *Edit – Properties*.

The View Property Editor form appears.

If you do not have write permission for the selected cell, the View Property Viewer appears instead of the View Property Editor. You can display but not edit the properties.



The screenshot shows the 'View Property Editor' dialog box. It has a title bar with 'View Property Editor' and a close button. The dialog is divided into three main sections: 'View', 'UNIX Permission Mode', and 'Properties'. The 'View' section contains fields for Name (schematic), Owner (empty), Group (1001), Last Modify (Sep 11 13:34:57), Read Path (tcases/VSE\_RAK/tutorial/EF\_PA\_ostg/schematic), and Write Path (tcases/VSE\_RAK/tutorial/EF\_PA\_ostg/schematic). The 'UNIX Permission Mode' section contains fields for Owner (rwx), Group (r-x), and Other (r-x). The 'Properties' section contains a list with 'dependency' and an 'EXPAND' button. At the bottom, there are buttons for OK, Cancel, Apply, Add, Delete, Modify, and Help.

The properties of the selected view appear at the bottom of the form.

3. In the View Property Editor form, you can add, delete, and modify view properties:

- ☐ To add a property, click *Add*.
- ☐ To delete a property, select the property and click *Delete*.

The selected property is removed from the form.

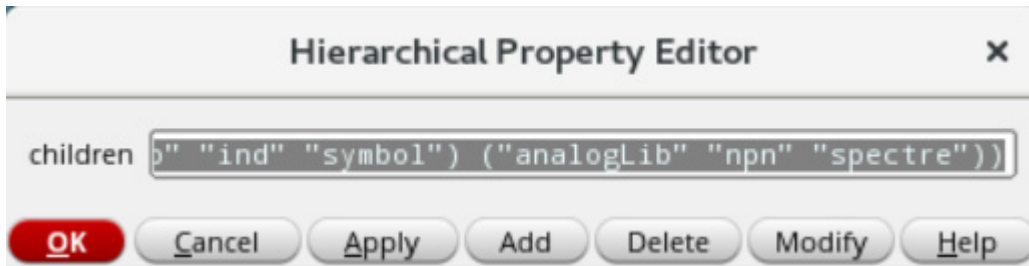
## Cadence Library Manager User Guide

### Library Management

---

- ❑ To modify a property, select the property and click *Modify*.

The dependency property shown above has an *Expand* button. If you click *Expand*, the Hierarchy Property Editor form appears. You can perform similar operations to those outlined above on this form.



4. Click *Apply*.

### ***Related Topics***

[View Property Editor Form](#)

[Adding Properties to a Library, Cell, or View](#)

[Modifying Properties of a Library, Cell, or View](#)

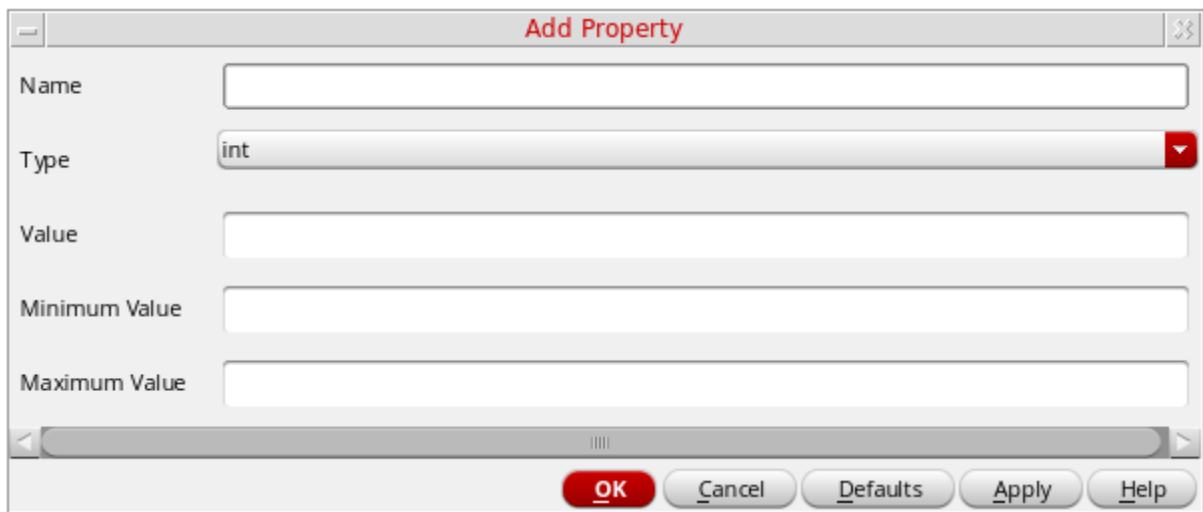
## Adding Properties to a Library, Cell, or View

You can add properties to a library, cell, or view from the Library/Cell/View Property Editor form that appears when you choose *Edit – Properties*.

To add properties:

1. On the Library/Cell/View Property Editor form, click *Add*.

The Add Property form appears.



2. In the *Name* field, type a name for the property.
3. In the *Type* drop-down list, select from the set of available property types.

The type you select determines the fields that appear on the Add Property form:

- ☐ If you select *int*, *float*, or *time* from the *Type* field, the subsequent fields are *Value*, *Minimum Value*, and *Maximum Value*.
- ☐ If you select *string* from the *Type* field, the subsequent fields are *Value* and *Possible Choices*.
- ☐ If you select *boolean*, *ILExpr*, *ILList*, *NLPEExpr*, *netSet*, *fileName*, or *hierProp* from the *Type* field, the subsequent field is *Value*.



## Cadence Library Manager User Guide

### Library Management

---

4. Type appropriate values in the remaining field or fields that appear based on the *Type* you selected in the previous step.

Type	Field	Valid Values
<i>int</i>	<i>Value</i> <i>Minimum Value</i> <i>Maximum Value.</i>	Any integer value  The values specified in the <i>Minimum Value</i> and <i>Maximum Value</i> fields define an inclusive range for a property value. The <i>Value</i> is the default value and must be in the specified range. You can specify <i>infinity</i> (no limit over 0), <i>-infinity</i> (no limit under 0), or leave the field blank to reflect no limit on the value.
<i>float</i>	<i>Value</i> <i>Minimum Value</i> <i>Maximum Value</i>	Any floating-point value
<i>time</i>	<i>Value</i> <i>Minimum Value</i> <i>Maximum Value</i>	Time and date values

## Cadence Library Manager User Guide

### Library Management

---

Type	Field	Valid Values
<i>string</i>	<i>Value</i>	Any string (no quotation marks) or no string
	<i>Possible Choices</i>	<p>Additional comma-separated list of strings (no quotation marks) or no strings</p> <p><b>Note:</b> You can force a double quotation mark to appear as part of a choice by typing a backslash in front of each quotation mark. For example:</p> <pre>\ "string\ "</pre> <ul style="list-style-type: none"> <li>■ If you leave the <i>Value</i> field blank, you must also leave the <i>Possible Choices</i> field blank; the result is that any string can be specified as a valid value for this property</li> <li>■ You may specify <i>Value</i> without specifying <i>Possible Choices</i>; the result is that <i>Value</i> is the default string and any other string can be specified as a valid value for this property</li> <li>■ If you specify strings in both the <i>Value</i> and the <i>Possible Choices</i> fields, the result is a a drop-down list containing the only valid values for the property with <i>Value</i> as the default selection</li> <li>■ You cannot leave the <i>Value</i> field blank and type a string in the <i>Possible Choices</i> field.</li> </ul>
<i>boolean</i>	<i>Value</i>	TRUE, true, t, yes, FALSE, false, nil, or no
<i>ILExpr</i>	<i>Value</i>	SKILL expression
<i>ILList</i>	<i>Value</i>	SKILL list
<i>NLPExpr</i>	<i>Value</i>	Expression evaluated by the netlister substitution language

## Cadence Library Manager User Guide

### Library Management

---

Type	Field	Valid Values
<i>netSet</i>	<i>Value</i>	.
<i>fileName</i>	<i>Value</i>	Any valid file name string
<i>hierProp</i>	<i>Value</i>	A valid list of properties

#### 5. Click *OK*.

The added property name and its value or values appear at the bottom of the Library/Cell/View Property Editor form as follows:

Value Set	Appearance
<i>Value</i> <i>Minimum Value</i> <i>Maximum Value</i>	<p>The value you specified in the <i>Value</i> field appears in an editable field to the right of the property name at the bottom of the Property Editor form; the value range defined by <i>Minimum Value</i> and <i>Maximum Value</i> appear between the property name and the field like this:</p> <pre>propertyName (minVal:maxVal) [_value_____]</pre> <p><b>Note:</b> <i>Value</i> must fall within the range defined by <i>Minimum Value</i> and <i>Maximum Value</i>, inclusive.</p>
<i>Value</i> and <i>Possible Choices</i>	<p>The appearance depends on what you typed in the fields as follows:</p> <ul style="list-style-type: none"> <li>■ If you left the <i>Value</i> field blank, a blank field appears to the right of the property name at the bottom of the Property Editor form</li> <li>■ If you specified a <i>Value</i> and left the <i>Possible Choices</i> field blank, the value appears in a field to the right of the property name at the bottom of the Property Editor form</li> <li>■ If you specified both a <i>Value</i> and <i>Possible Choices</i>, these values appear in a drop-down list to the right of the property name at the bottom of the Property Editor form with <i>Value</i> as the default selection</li> </ul>
<i>Value</i> alone	<p>The value appears in an editable field to the right of the property name at the bottom of the Library/Cell/View Property Editor form</p>

***Related Topics***

[Add Property Form](#)

[Editing Library Properties](#)

[Editing Cell Properties](#)

[Editing View Properties](#)

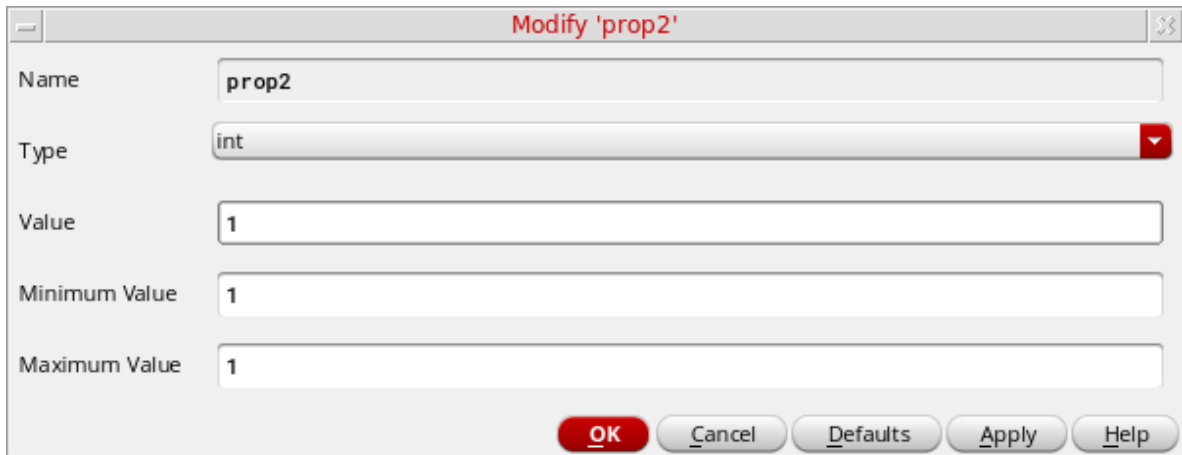
## Modifying Properties of a Library, Cell, or View

You can modify properties of a library, cell, or view from the Library/Cell/View Property Editor form that appears when you choose *Edit – Properties*.

To modify properties:

1. On the Library/Cell/View Property Editor form, click *Modify*.

The Modify '*propertyName*' form appears.



The name of the selected property appears in the title of the form.

2. Make your desired changes on the form.
3. Click *OK*.

### ***Related Topics***

[Modify '\*propertyName\*' Form](#)

[Adding Properties to a Library, Cell, or View](#)

[Editing Library Properties](#)

[Editing Cell Properties](#)

[Editing View Properties](#)

## Updating a Managed File

Changes made to a design or component are not visible to members of the design team in a managed design until the design or component is checked in. If you need to use the latest version of an item (for example, a view or the base level CDF properties of a view) checked out to another designer:

1. On the Library Manager form, select an item (library, cell, or view).
2. Choose *Design Manager – Update*.

The Update Library, Update Cell, or Update View form appears.

This form runs the `gdmupdate` command.

3. To pass a string to the `-name` argument of the `gdmupdate` command:
  - a. Select the *Update From* check box.
  - b. In the *Update From* field, type a valid tag specification or TDM release name.
4. To pass a string to the `-extra` argument of the `gdmupdate` command:
  - a. Select the *Use Options* check box.
  - b. In the *Use Options* field, type a valid string for the `-extra` argument.

The software checks the `cds.lib` file to ensure it has the latest information and then reads the latest edits made to a view or the base level CDF properties of a view into virtual memory, even if the item is checked out to another user. Status messages appear in the *Messages* scrolling area at the bottom of the form.

This command also redraws opened designs affected by edits to a view or its properties.

### ***Related Topics***

[Library Manager Form](#)

[Generic Design Management \(GDM\) Commands](#)

## Library Display Settings

You can customize the display of libraries in the Library Manager. For example, you can specify that certain libraries be hidden or displayed in a different color or with a particular icon. You do this by setting attributes on libraries.

Cadence provides a set of predefined attributes. You can also add custom attributes.

Attributes are saved in `displayPrefs` files. The Library Manager displays all attributes defined in any `displayPrefs` file found by the Cadence Search File mechanism (CSF). All directories listed in your `setup.loc` file are searched for a `.cadence/libManager/displayPrefs` file. The directories are read in the reverse order in which they are listed in the `setup.loc` file. If an attribute is defined multiple times, the value from the last definition read is used.

The Library Manager's Display Settings form enables you to view the list of existing attributes (predefined as well as custom), modify the display settings of these attributes, and define new attributes. To define an attribute, you specify an attribute name and choose the display settings, such as a specific color or icon, that applies to all libraries tagged with that attribute. Any changes you make are always saved to the `current_working_dir/.cadence/libManager/displayPrefs` file.

You can then set any of these attributes on libraries. To set an attribute on a library, you need to edit your library definition file and add an `ASSIGN` statement for the library.

### ***Related Topics***

[Creating New Library Attributes](#)

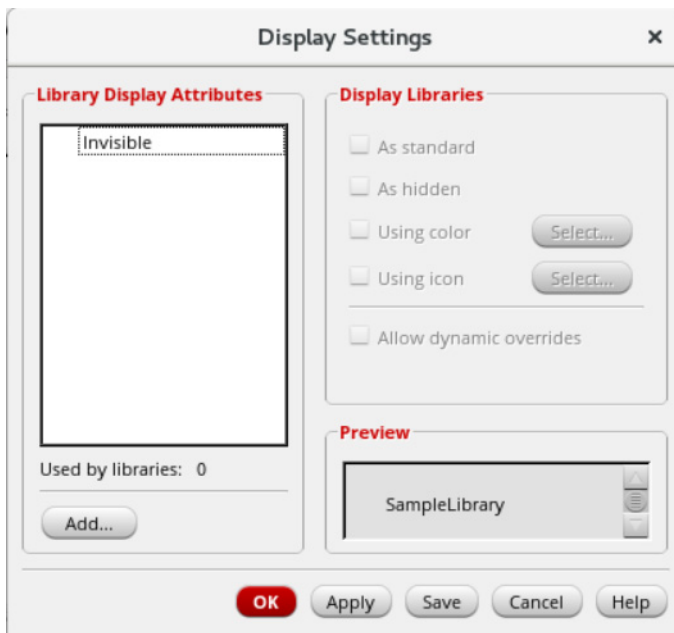
## Setting Display Options for Libraries

Cadence recommends that you never edit the `displayPrefs` file manually.

To set display options for libraries:

1. Select *Edit - Display Settings*.

The Display Settings form appears.



The *Library Display Attributes* list box displays all predefined and custom attributes. It also displays any attributes that have been set on libraries in your `cds.lib` file but that have not yet been defined in a `displayPrefs` file.

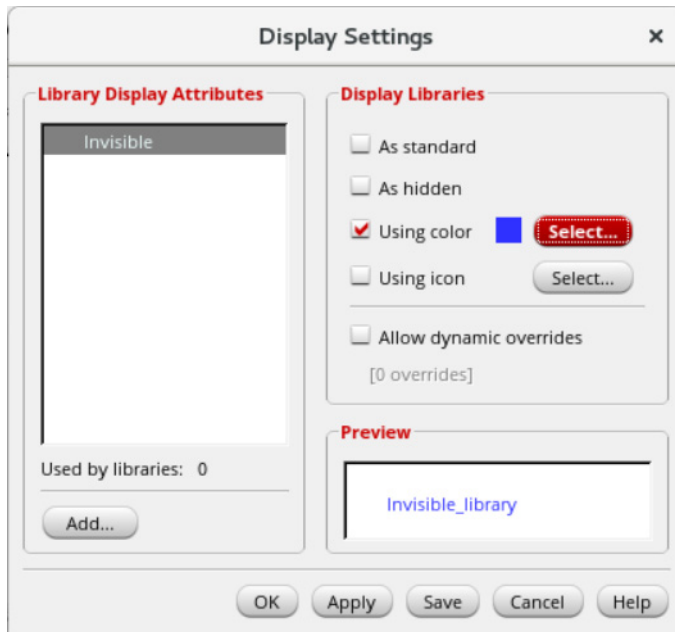
To create a new attribute, see [Creating New Library Attributes](#).



## Cadence Library Manager User Guide

### Library Management

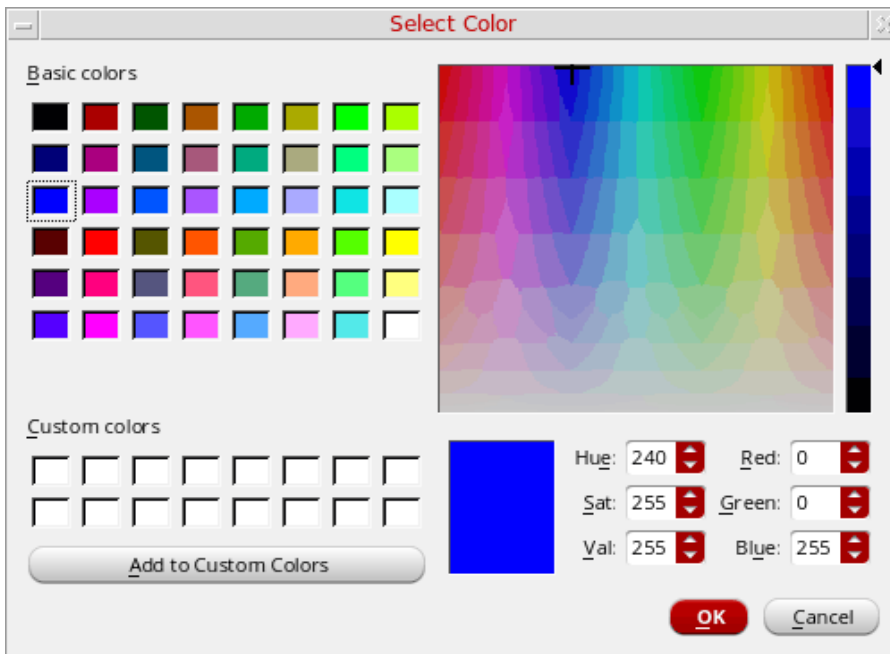
#### 2. Select an attribute.



The *Matching libraries* field under the *Library Display Attributes* listbox displays the number of libraries on which the selected attribute is currently set.

#### 3. Select the new display settings for the attribute.

#### 4. Click *Select* to select a color from the Select color dialog box.



## Cadence Library Manager User Guide

### Library Management

---

5. Choose a color from the *Basic colors* or *Custom colors* table, or create a custom color by moving the cross in the color spectrum to the color you want and then clicking *Add to Custom Colors*.
6. The color you select is displayed next to the *Using color* option.  
The icon you select is displayed next to the *Using icon* option.
7. The *Preview* field shows a sample library name with the display settings that you have selected.
8. Click *OK/Apply/Save*.

*Save* applies your changes and also saves them to the `displayPrefs` file immediately.

Libraries tagged with the attributes you changed are now displayed in the new settings.

The new or modified attribute definitions are saved in the `current_working_dir/.cadence/libManager/displayPrefs` file. If this file does not exist, it is created. If you clicked *Save*, your changes are saved to this file immediately; if you clicked *OK* or *Apply*, the file is updated when you exit the Library Manager.

### ***Related Topics***

[Display Settings Form](#)

[Selecting an Icon for a Library Display Attribute](#)

[Setting Attributes on a Library](#)

## Selecting an Icon for a Library Display Attribute

You can select icons for library display attributes. The Library Manager uses the Cadence Search File mechanism (CSF) to find icons—both Cadence application icons as well as any custom icons that you or your site have added.

Specifically, the Library Manager looks for the following two directories:

```
icons/library/16x16  
icons/16x16
```

For every location in the `setup.loc` file, the following subdirectories are searched:

```
.cadence  
.  
(the exact location)  
cdssetup
```

Also, from every location, the first subdirectory that contains an `icons` directory is used. For example, if an `icons` directory is found in `locationA/.cadence`, then `locationA` and `locationA/cdssetup` are not searched.

If multiple definitions are found for an icon, that is, a file of the same base name is found in multiple locations, the definition from the location that has higher precedence in the `setup.loc` file is used, as per CSF rules.

If you create custom icons, place them in the following sub-directory of any directory that is listed in your `setup.loc` file, such as `$HOME`:

```
icons/library/16x16
```

If an icon is defined in both an `icons/16x16` directory and an `icons/library/16x16` directory, the definition in the `icons/library/16x16` directory is used.

To select an icon for a library display attribute,

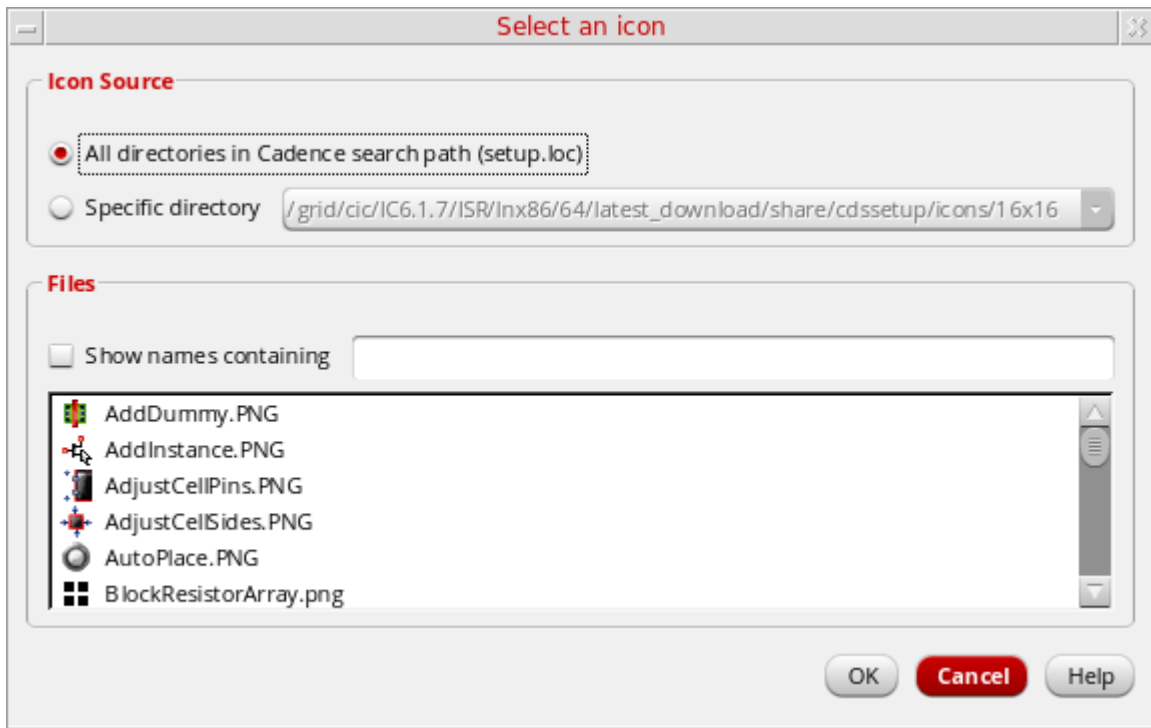
1. In the Display Attribute form, select an attribute.
2. In the *Display Libraries* section, select *Using Icon*, then click *Select*.

## Cadence Library Manager User Guide

### Library Management

---

The Select an icon form appears.



3. In the form's *Icon Source* field, select the directories from which you want to display icons in the *Files* list.

Until you add custom icons to other locations in your `setup.loc` file, the only directories listed in this field are those containing Cadence application icons, such as the `your_install_dir/share/cdssetup/icons/16x16` icon directory.

4. Select *Show names containing* and specify a pattern to filter icon file names.

Only the file names containing the pattern are displayed. For example, `ibr` displays only those file names that contain `ibr`, such as `NewLibrary.png`.

5. From the *Files* list, select the icon that you want to use for the attribute.

You can place your cursor over an icon file name to view the directory from which it is obtained.

6. Click *OK*.

The icon you selected is displayed next to the *Using Icon* field in the Display Settings form. The *Preview* field also displays the icon next to the sample library name.

## Cadence Library Manager User Guide

### Library Management

---

#### ***Related Topics***

[Select an icon Form](#)

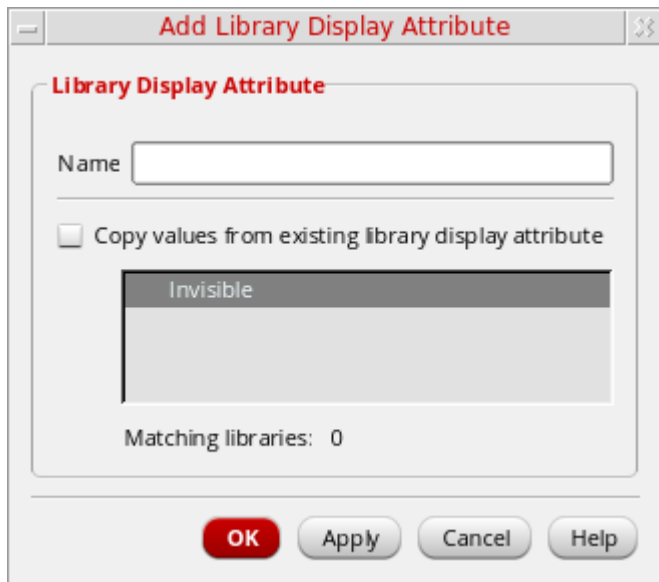
## Creating New Library Attributes

You can create new library display attributes and define the display settings for them. Attributes that you add are saved in the `current_working_dir/.cadence/libManager/displayPrefs` file. You can add any number of attributes. Cadence recommends that you never edit the `displayPrefs` file manually.

To create new library display attributes,

1. Select *Edit – Display Settings*.
2. In the Display Settings form, in the *Library Display Attributes* section, click *Add*.

The Add Library Display Attribute form appears.



3. In the *Name* field, specify a name for the attribute. Names cannot include spaces.
4. To copy display settings from another attribute, select *Copy values from existing library attribute*, then select the attribute from the available list.
5. Click *OK/Apply*.

The new attributes are listed in the *Library Display Attributes* listbox in the Display Settings form.

6. If you did not copy display settings from another attribute, select them in the *Display Libraries* section of the Display Settings form.
7. Click *OK/Apply/Save*.

## Cadence Library Manager User Guide

### Library Management

---

The new attributes are saved in the

`current_working_dir/.cadence/libManager/displayPrefs` file. If the file does not exist, it is created. If you clicked *Save*, your changes are saved to this file immediately; if you clicked *OK* or *Apply*, the file is updated when you exit the Library Manager.

To customize attributes on a per-site or per-project basis, copy the `displayPrefs` file to the appropriate directories. The Library Manager uses CSF search to find attributes.

For more information, see [Library Display Settings](#).

### ***Related Topics***

[Setting Display Options for Libraries](#)

[Library Display Settings](#)

[Setting Attributes on a Library](#)

## Setting Attributes on a Library

To set an attribute on a library, in your `cds.lib` file:

```
ASSIGN libName DISPLAY attributeName
```

where *libName* is the library on which you want to set the attribute and *attributeName* is the name of the attribute.

The library must be already defined with the `DEFINE` statement earlier in the file, otherwise the `ASSIGN` statement is ignored.

The next time you start the Library Manager, the library appears in the display settings specified for the attribute (for example, in a specific color).

In the `cds.lib` file, you can also set a new attribute that is as yet undefined in a `displayPrefs` file. The next time you start the Library Manager, this new attribute appears in the *Library Attributes* list in the Display Settings form. You can then define the display options for the attribute.

### ***Related Topics***

[Setting Display Options for Libraries](#)

[Viewing the cds.lib Updates](#)



## Overriding Customized Library Display Settings

The Display Options form lets you override any of the custom display settings that you have set on libraries. Overrides apply to the current session only.

To override custom library display settings,

1. Select *View – Display Options*.

The Display Options form appears.

**Display Options**

**For Objects**

Library View

**Display Overrides**

- ☐ Show all libraries using standard style
- ☐ Show hidden libraries
- ☒ Show library colors
- ☒ Show custom library icons
- ☒ Show Lists view library icons

**State Analysis**

- ☒ Enable query of Design Management states
  - ☐ Enable delay before DM syncs
    - Wait 2.5 seconds after selecting
- ☒ Enable poll
  - Interval 30 seconds
  - Idle limit over interval 10 seconds

**Resources**

Custom library display attributes Edit...

OK Apply Cancel Help

2. In the *Library* tab *Display Overrides* section, select or deselect any of the available options.
3. In the *View* tab, the only option in the *For Objects* section is *Show extended states*. This option allows DM tables to be shown in any viewing mode.

## Cadence Library Manager User Guide

### Library Management

---

It should be used along with the *Enable query of Design Management states* option in the Library page.

4. In the *State Analysis* section, select or deselect any of the available options.
5. To edit library display settings, in the *Custom library display attributes* field, click *Edit*.

The Display Settings form appears.

The screenshot shows the 'Display Settings' dialog box. It has a title bar with 'Display Settings' and a close button. The dialog is divided into three main sections: 'Library Display Attributes', 'Display Libraries', and 'Preview'. The 'Library Display Attributes' section on the left contains a large rectangular area with a dashed border and the word 'Invisible' in the top-left corner. Below this area, it says 'Matching libraries: 0' and has an 'Add...' button. The 'Display Libraries' section on the right has four checkboxes: 'As standard', 'As hidden', 'Using color', and 'Using icon'. The 'Using color' and 'Using icon' checkboxes are accompanied by 'Select...' buttons. The 'Preview' section at the bottom right shows a rectangular area with the text 'SampleLibrary'. At the bottom of the dialog, there are five buttons: 'OK' (highlighted in red), 'Apply', 'Save', 'Cancel', and 'Help'.

### ***Related Topics***

[Display Options Form](#)

[Display Settings Form](#)

[Library Display Settings](#)

## Creating and Displaying a Combined Library

The Library Manager allows you to group a set of libraries and display them as a *combined* library.

A combined library is a virtual library made up of other libraries. The data is not physically moved or copied; it is just displayed in consolidated form under the combined library.

This feature helps you manage the display of libraries for your needs by letting you group together libraries for a specific purpose or to reduce the number of libraries displayed in the *Library* list box. For example, if you use a base set of libraries with some additional libraries for one process and with another set of libraries for another process, you could group the relevant libraries together for each process.

### Creating Combined Libraries

Combined libraries are created by setting an `ASSIGN` statement with a `COMBINE` attribute in your `cds.lib` file.

To create a combined library

1. Create a new directory for the combined library.

The combined library must be valid and defined before the `ASSIGN` statement for it to work properly.

2. In your `cds.lib`, add the following statement:

```
ASSIGN combinedLibName COMBINE libA libB ...
```

where *combinedLibName* is the name of the top-level library and *libA* and *libB* are the libraries that comprise the combined library.

The combined library must be valid and have a physical representation, even if it is an empty directory. Any invalid libraries are ignored in libManager. Invalid libraries present in the `ASSIGN` statement generates a warning message.

3. Ensure that a `DEFINE` statement for the new combined library and for all the libraries that are combined appears before the `ASSIGN` statement in `cds.lib`.

All the libraries specified in the statement must already be defined with the `DEFINE` statement earlier in the file, otherwise the invalid libraries are ignored by Library Manager.

A library can be placed in more than one combined library.

## Cadence Library Manager User Guide

### Library Management

---

With the `ASSIGN` statements, you can build up a hierarchy of libraries.

For example:

```
DEFINE analogLib /home/libs/analogLib
DEFINE sbaLib /home/libs/sbaLib
DEFINE demoLib /home/libs/demoLib
DEFINE newLib /home/libs/newlib
DEFINE testLib /home/libs/testLib

ASSIGN newLib COMBINE analogLib sbaLib
ASSIGN testLib COMBINE newLib demoLib
```

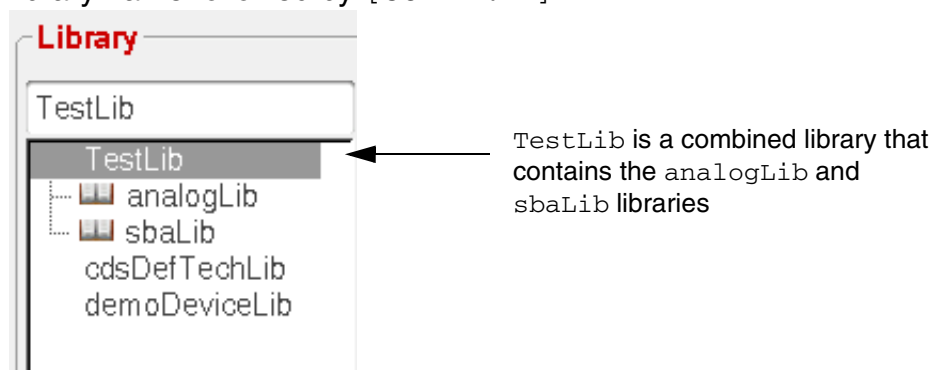
## Display Combined Libraries

In the Library Manager, combined libraries are displayed in the *Library* list just like any other library.

Combined libraries are displayed in a tree form.

A + icon next to a library name indicates that it is a combined library and has a hierarchy under it. Double-clicking on the library or clicking the + icon displays the libraries it contains. The individual libraries that comprise a combined library are not displayed at the top-level, they are only displayed under the combined library.

Tooltips for combined libraries also indicate that they are combined: the tooltip displays the library name followed by `[COMBINED]`.



When a combined library is selected in the *Library* list, the other list boxes display composite data from all the libraries that comprise the combined library.

The *Cell* list box displays all the cells from all the libraries under the combined library. The tooltip for each cell shows the same library to which it belongs.

## Cadence Library Manager User Guide

### Library Management

---

Similarly, the *Files in Library* list box displays library-level files from all the libraries in the combined library. Files that are found in more than one library are listed with the library name displayed in brackets after the file name to differentiate them. Tooltips display the origin of the other files.

When you select a cell in the *Cells in Library* list box, its views and files are displayed as usual. Tooltips indicate the library and cell information for the view or file.

However, if a cell of the same name is found in two libraries, consolidated data for both cells is displayed when you select the cell. The *Views in Cell* list box displays the views found in both cells, with the library name in brackets to differentiate them. Similarly, the *Files in Cell* list box displays cell-level files for both cells, with the library name in brackets. Also, the tooltip for the cell indicates that the cell is combined, for example: `TestLib/n2port [COMBINED]`.

For combined libraries, categories are also merged—the *Category* list box displays all categories for all libraries in the combined library. Selecting a category that is common to two libraries displays data from both libraries that belongs to that category.

**Note:** Libraries are combined for display purposes only. Any edit commands, such as modifying categories, copying, or renaming libraries or cells apply to the physical library only, not to the combined library. For example, if you copy a combined `libraryA` that contains its own data as well as `libraryB` and `libraryC`, only the contents of `libraryA` are copied to the new library. The forms for these commands also display information about the physical library only. The only exception to this is the *Delete* command, which displays information about the combined data and gives you the option of deleting from multiple locations, if applicable. See the descriptions of the *Delete* commands earlier in this chapter for more information.

You cannot copy or rename a combined cell (a cell that is found in multiple libraries in a combined library) unless you select a specific view or select the cell from the library to which it belongs.

### ***Related Topics***

[Library Manager Form](#)

[Viewing the cds.lib Updates](#)

# **Cadence Library Manager User Guide**

## **Library Management**

---

---

## Library Browsing

---

The Library Browser form is similar to the Library Manager form, but it does not have menus and is used for displaying and selecting cellviews only. Some applications use the Library Browser form; others use the Library Manager. The Library Browser form appears when you click the *Browser* button in a Virtuoso form.

You can use the Library Browser form to perform the following tasks:

- Display and select libraries, categories, cells, and views specified in your `cds.lib` file.
- Filter libraries, categories, cells, and views to narrow your selection

### ***Related Topics***

[Opening the Library Browser Form](#)

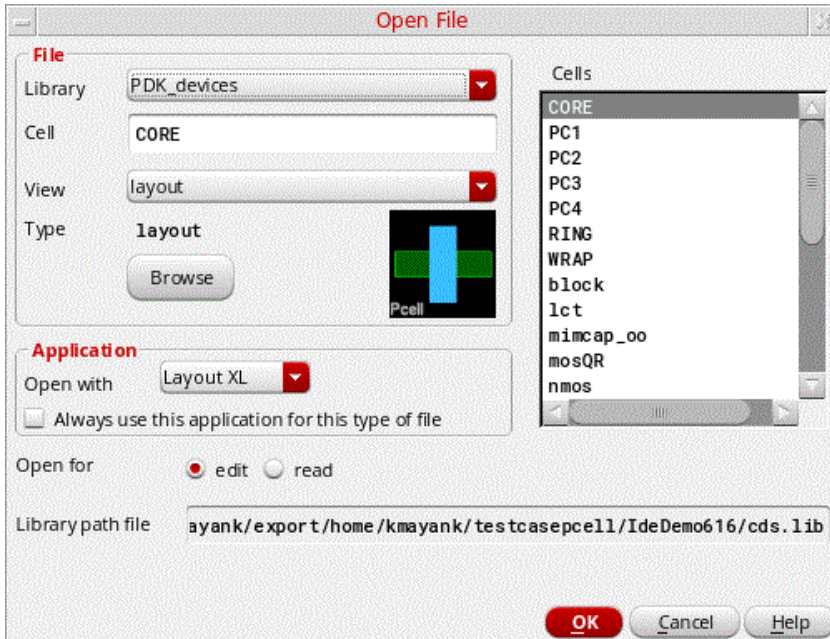
[Selecting a View Using the Library Browser](#)

[libSelectCellViewCombineMode](#)

## Opening the Library Browser Form

To open the Library Browser form manually:

1. Click *Browse* on a form. For example, click *Browse* on the Open File form.



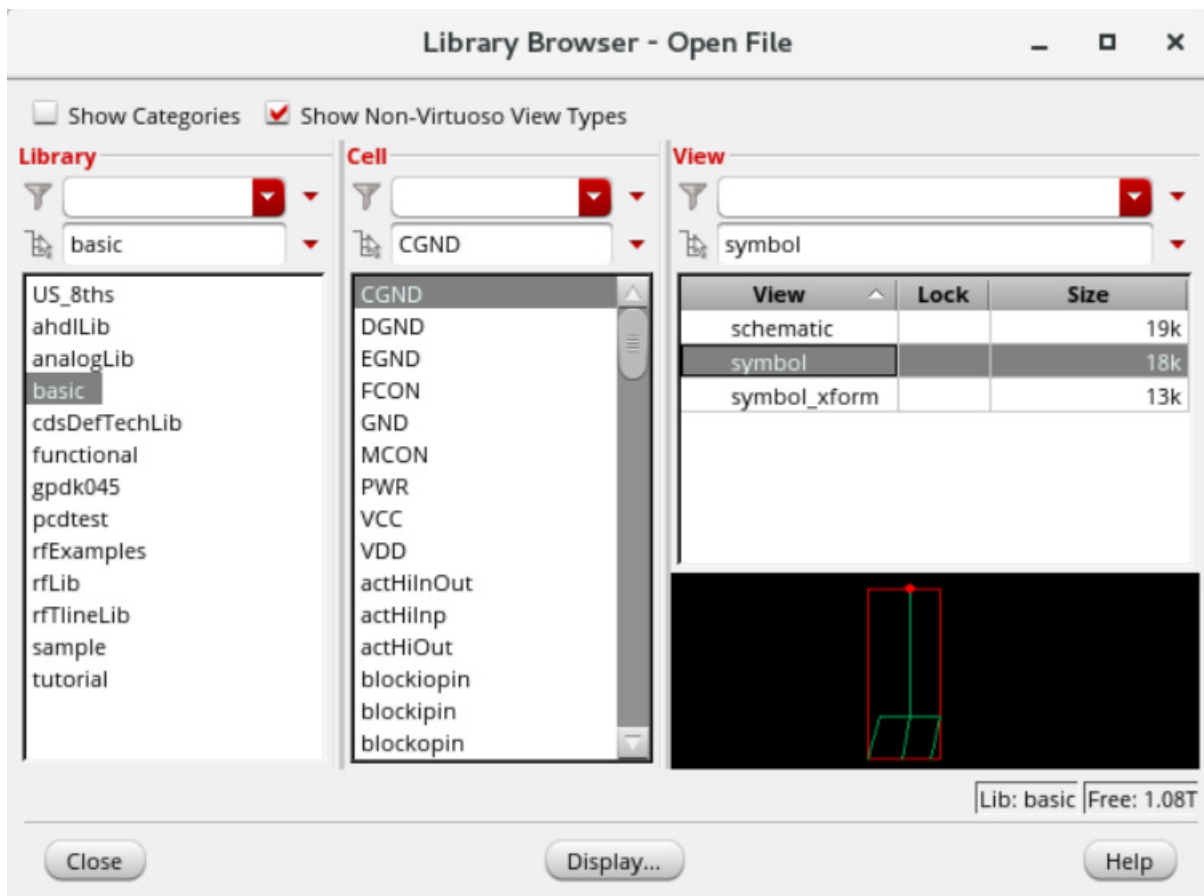
2. The cells in the *Cells* list box of the Open File form are displayed in alphanumeric order.
3. To use the same sort order as used in Library Manager, set `designEditor.fileSpec`.



## Cadence Library Manager User Guide

### Library Browsing

4. The Library Browser form appears.



### ***Related Topics***

[Library Browser Form](#)

[showNonVirtuosoViewtypes](#)

## Setting the Library Browser Form To Open Automatically

To set the Library Browser form to open automatically from specific forms, follow these steps:

1. From the CIW, choose *Options – File Preferences*.

The File Preferences form appears.

The screenshot shows the 'File Preferences' dialog box. It has a title bar with the text 'File Preferences'. Inside, there are three main sections. The first section, 'File Preference', contains 'Recently Used File List' with a value of '10' and 'Entries', and 'Auto Save Database After' with a value of '500' and 'Edits'. The second section, 'Open Browser Automatically For', contains three radio button options: 'Open Form', 'Create Instance Forms', and 'Other Forms', each with 'yes' and 'no' choices. All three are currently set to 'no'. The third section, 'CIW Preferences', contains 'Prompt On Exit' with 'yes' and 'no' choices, where 'yes' is selected. At the bottom of the dialog are four buttons: 'OK' (highlighted in red), 'Cancel', 'Defaults', and 'Help'.

2. In the *Open Browser Automatically For* group box, select *yes* for the forms from which you want the Library Browser form to open automatically.

You can select any or all form types.

3. Click *OK*.

The next time you open any of the selected types of forms, the Library Browser form appears automatically. You do not have to restart the software for your preferences to take effect.

### ***Related Topics***

[Library Browser Form](#)

## Selecting a View Using the Library Browser

To select a cellview using the Library Browser form, follow these steps:

1. In the *Library* list box, select a library.

The cells contained in that library appear in the *Cell* list box.

2. To display the views associated with a cell, select the cell name.

The names of the views for that cell appear in the *View* list box.

3. In the *View* list box, double-click a view name.

The Library Browser form closes. The selected library, cell, and view names appear in the appropriate fields on the parent form, from where you opened the Library Browser form.

### ***Related Topics***

[Library Browser Form](#)

# **Cadence Library Manager User Guide**

## Library Browsing

---

---

## Data Copying

---

Use Library Manager copy functions to assemble design and reference libraries by copying cells or views from libraries, that are specified in your `cds.lib` file, into other libraries.

Function	Description
<i>Copy</i>	Lets you copy a library, cell, view, or file.  Opens the Copy Library, Copy Cell, Copy View, or Copy Library File form.
<i>Copy Wizard</i>	Lets you copy a library, cell, or view; copy hierarchically; copy by view; or copy by configuration.  Opens a dynamic form that lets you choose how you want to copy.

The Library Manager only lets you edit and manage OpenAccess libraries, any library that has a library-level `prop.xx` file is grayed-out and cannot be copied. Cell-level `prop.xx` files are not copied when you copy the cell. Copy commands can also fail if you have `prop.xx` files.

### ***Related Topics***

[Copy Function in the Library Manager](#)

[Selecting Text in the Copy Wizard](#)

[Editing Text in the Copy Wizard](#)

[Setting Copy and Rename Preferences](#)

## Copy Function in the Library Manager

The Copy function opens the Copy Library, Copy Cell, Copy View, Copy Library File, or Copy Cell File form depending on what you have selected on the Library Manager form.

- Copy Library lets you copy a single library to a new library name Copy Cell lets you copy a cell to a new cell name or into another library. You also have the option to copy cells hierarchically.
- Copy View lets you copy a view to a new view name or into another cell or library . You also have the option to copy views hierarchically.
- Copy Library File lets you copy a single library file to a new name or a new library.
- Copy Cell File lets you copy a single cell file to a new name, a new cell, or a new library.

### ***Related Topics***

[Copying a Library in the Library Manager](#)

[Copying a Cell in the Library Manager](#)

[Copying a Cell Hierarchy in the Library Manager](#)

[Copying a View in the Library Manager](#)

[Copying a View Hierarchy in the Library Manager](#)

[Copying a Library File](#)

[Copying a Cell File in the Library Manager](#)

## Pre-Copy Checks in the Library Manager

Certain checks are performed prior to copying data. The Library Manager invokes any pre-copy checks defined by applications.

To ensure technology database compatibility between source and destination libraries, the following checks are performed:

- Ensure that the names of files to be copied do not have a space in them. Files with names that have spaces does not get copied.
- If the source library contains a local technology database file (`tech.db` file), then check if the destination library associates with a technology database, either through attachment or by containing a `tech.db` file in the library. If the destination library associates with a technology database, then the copy command is aborted.

If the above check succeeds, then the following additional checks are performed for compatibility of design data:

1. Compatibility Check: Check for compatibility between the technology databases of the source and destination library and do the following:
  - ❑ If the source and destination technology databases are equal, then proceed to the existence check.
  - ❑ If the source technology database is a subset of the technology database of the destination library, then proceed to the existence check.
  - ❑ If there are any conflicts with a severity level of error between the technology databases, then abort the copy command. The following table lists the checking criteria:

Technology objects	Checking criteria for compatibility
oaLayers (oaPhysicalLayers / oaDerivedLayers)	Same name then same number, same number then same name
oaPurposes	Same name then same number, same number then same name
oaStdViaDefs	Same viaDef name, two compatible layers and a cutLayer in the viaParameter
oaCustomViaDefs	Same viaDef name and two compatible layers
oaScalarSiteDefs	Same name, same width, and same height

## Cadence Library Manager User Guide

### Data Copying

---

#### Technology objects

#### Checking criteria for compatibility

oaArraySiteDefs

No compatibility checking; check that there is no siteDef of the same name but a different type

dbuPerUU (technology attribute)

Same value

For view types of dbcMaskLayout, dbcSchematic, dbcSchematicSymbol, and dbcNetlist

userUnits (technology attribute)

Same unit name

For view types of dbcMaskLayout, dbcSchematic, dbcSchematicSymbol, and dbcNetlist

2. Existence Check: If the source technology database and the destination technology database are compatible, then check for the existence of technology objects in the destination technology database. If there are any missing technology objects for a cellview, that cellview is still copied and warning messages are issued.

Information about errors is displayed in the CIW. Modify your source data accordingly and then try to copy it again.

#### ***Related Topics***

[Copy Function in the Library Manager](#)

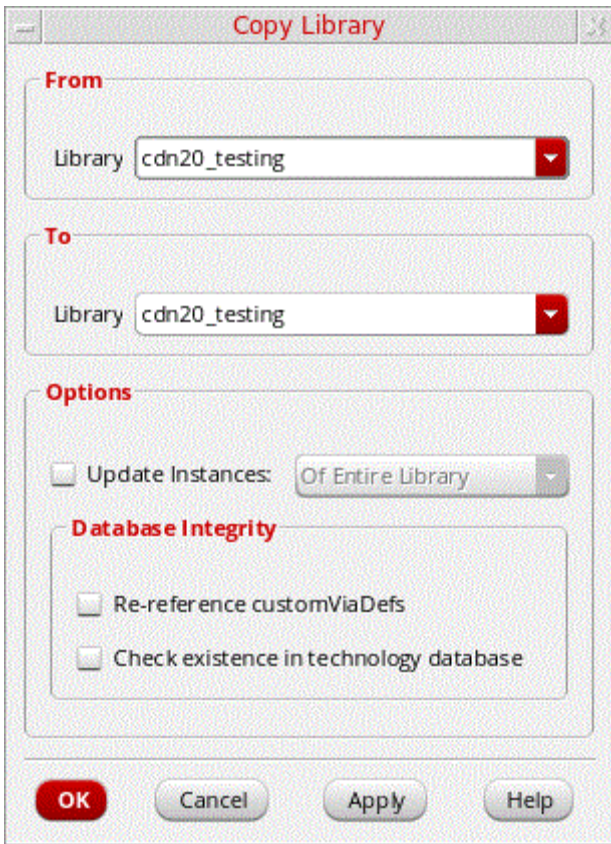


## Copying a Library in the Library Manager

To copy a library to another name, follow these steps:

1. In the *Library* list box, select a library.
2. Choose *Edit – Copy*.

The Copy Library form appears.



3. In the *To Library* field, type a destination library name or select a library name from the drop-down list. You can type a new name or select an existing library.
4. Select the *Update Instances* check box to update the cells and views in the destination library with the new library name.

When the *Update Instances* check box is not selected, the software leaves references to the *From Library* name unchanged.

For example, all instances of `.../oldLib/NAND/symbol` continue to reference the original library and remain `.../oldLib/NAND/symbol`.

## Cadence Library Manager User Guide

### Data Copying

---

5. Select the options in the *Database Integrity* field if you want to update and validate technology data in the destination library after the copy command is completed.
6. Click *OK*.

If the destination library already contains the `tech.db` file, the source library's `tech.db` is not copied. Otherwise, while copying a source library to an existing library, the `tech.db` file associated with the source library also gets copied to the destination library.

In case you selected an existing library, the Copy Problems form appears.

When you copy a source library to a destination library using Library Manager, the entire library is copied (including `data.dm`), irrespective of how the variables of `addPropFiles`, `addLibPropFiles`, and `addCellPropFiles` are set.

### Copying to a New Library

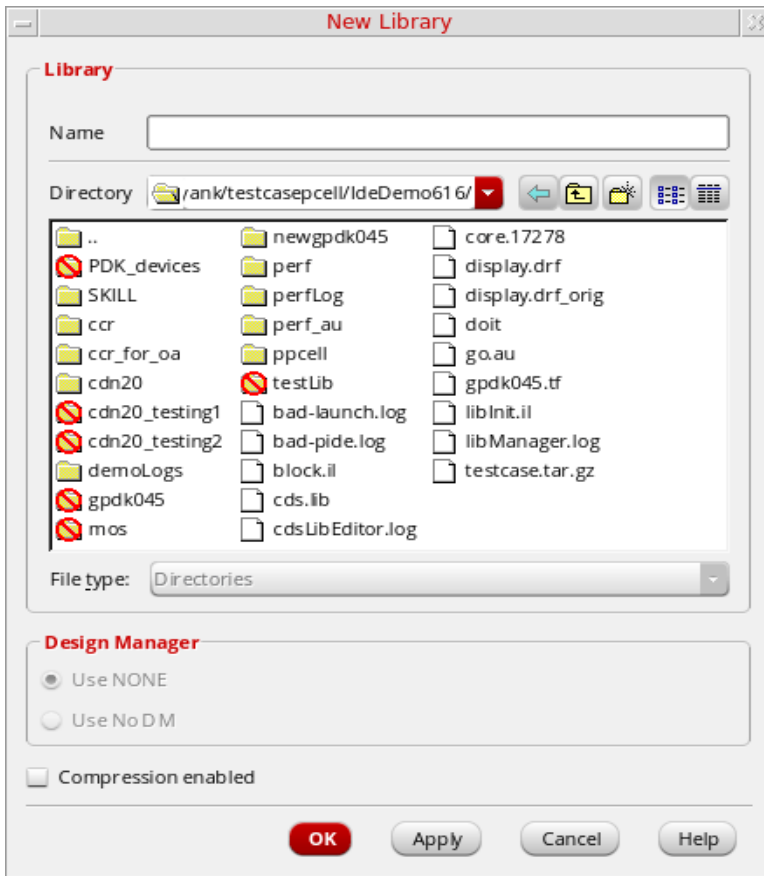
When you are copying to a new library and you click *OK* on the Copy Library form, the New Library form appears. To complete the copy operation to a new library, do the following:

## Cadence Library Manager User Guide

### Data Copying

---

1. On the New Library form, use the *Directory* navigation tools (list box and toolbar buttons) to specify the destination directory into which you want to copy the new library. You can also type a directory path in the *Directory* field.



2. In the *Design Manager* group box, specify a design management option.

If there is a design management system available to you and you choose *Use No DM* now, you can still decide to check it in later on.

3. Click *OK*.

### ***Related Topics***

[Copy Library Form](#)

[New Library Form](#)

[Resolving Copy Problems](#)

## Resolving Copy Problems

When copying to an existing library, you click *OK* on the Copy Library form, the Copy Problems form appears listing the copy problems in a sortable table.

Destination Library: amsPLL

One or more of the following problems have occurred:

1. The source file does not exist.
2. A destination file will be overwritten.
3. A destination file is checked out.
4. A destination file is opened for edit.
5. A destination file conflicts with another.
6. A source library file is not valid in a different library.
7. A source OpenAccess cellview master file has zero size.

From Library ▾	From Cell	From View	To Cell	To View	Error	Action
basic	/	data.dm	/	data.dm	Would Overwrite	Don't Copy
analogLib	/	data.dm	/	data.dm	Would Overwrite	Don't Copy

Buttons: OK, Fix Errors, Overwrite All, Cancel, Help

To resolve copy problems, do the following:

1. In the *Error* column for each item, view the error.
2. In the *Action* column for each item, select an action.

For example, the following actions are available for the *Would Overwrite* error:

- ☐ *Don't Copy*: Does not copy the item to the *Destination Library*.
  - ☐ *Overwrite*: Writes the item to the *Destination Library*, overwriting the item of the same name.
  - ☐ *Auto Rename*: Appends the *From Cell* name (if different) and the *From Library* name (if different), to the final cell name by adding an underscore prefix. If a destination cell with the same name already exists, an underscore and a number (for example, *\_01*) is automatically added to the name.
3. Select one of the following options as an action for all items:
    - ☐ *Fix Errors* selects the software's choice for fixing the problem for all items (for example, using *Auto Rename* to fix the *Would Overwrite* error).
    - ☐ *Overwrite All* selects *Overwrite* for all items.

## Cadence Library Manager User Guide

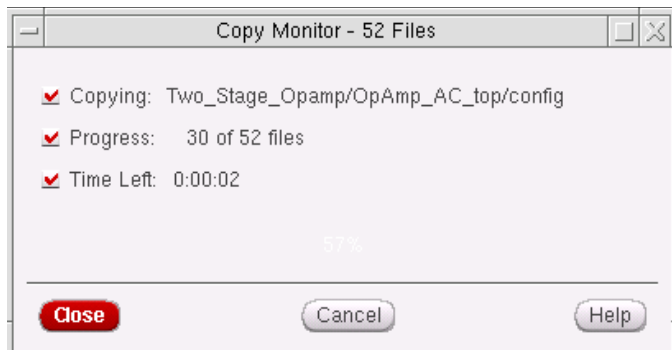
### Data Copying

---

You can click *Fix Errors* or *Overwrite All* first and then select a different action for some individual items.

#### 4. Click *OK*.

The Copy Monitor status window is displayed.



The copy operation completes using the specified actions.

### ***Related Topics***

[Copying a Library in the Library Manager](#)

## Copying a Cell in the Library Manager

To copy a cell to another name or into another library, follow these steps:

1. Select a cell.
2. Choose *Edit – Copy*.

The Copy Cell form appears.

The screenshot shows the 'Copy Cell' dialog box. It has a title bar 'Copy Cell'. Below it are two groups: 'From' and 'To'. Each group has a 'Library' dropdown menu and a 'Cell' text field. In the 'From' group, 'Library' is 'analogLib' and 'Cell' is 'MOS\_a2d'. In the 'To' group, 'Library' is 'analogLib' and 'Cell' is 'MOS\_a2d'. Below these are 'Options'. The 'Options' section has a 'Copy Hierarchical' checkbox (checked), a 'Skip Libraries' checkbox (checked), a text area containing a list of libraries (basic, cdn20, cdn20\_abutment, cdn20\_testing, cdsDefTechLib, sample, smgAdvancedMSLib, smgBasicMSLib), an 'Edit' button, a radio button for 'Exact Hierarchy' (selected), an 'Extra Views' section with 'Names' and 'Types' fields, a 'Views To Copy' section with a radio button for 'All Views' (checked), an 'Update Instances' dropdown menu (set to 'Of Entire Library'), a 'Database Integrity' section with two unchecked checkboxes ('Check existence in technology database' and 'Re-reference customViaDefs'), and an 'Add To Category' section with a dropdown menu (set to 'Cells'). At the bottom are 'OK', 'Cancel', 'Apply', and 'Help' buttons.

The selected library and cell appear in the *Library* and *Cell* fields in the *From* group box. They also appear in the *Library* and *Cell* fields in the *To* group box. You can change any or all of these values.

You can set the maximum cell name length allowed by using the `CDS_MAX_CELL_NAME_LENGTH` environment variable.

3. To complete the copy cell operation, follow the steps for one of the following tasks:
  - ❑ Copying a cell to another name in the same library

To copy a cell to another name in the same library,

- a. Follow the steps mentioned above.
- b. In the *To* group box in the *Cell* field, type a destination cell name.
- c. Click *OK*.

The *From* cell name is copied to the *To* cell name in the same library.

☐ Copying a cell to another library

To copy a cell to another library,

- a. Follow the steps mentioned above.
- b. In the *To* group box in the *Library* field, type or select a destination library name.
- c. You can type a new library name or select an existing library from the drop-down list.
- d. In the *To* group box in the *Cell* field, type a destination cell name.
- e. If you do not change the cell name in the *To* group box, the copied cell displays the same name as the original cell.
- f. Click *OK*.

The *From Cell* is copied to the *To Cell* in the *To Library*. If the destination library does not already exist, the New Library form appears, where you can specify a location and design management option for the new library.

### ***Related Topics***

[Copy Cell Form](#)

[Copying to a New Library](#)

[Copying a Cell Hierarchy in the Library Manager](#)

[Updating Cell Instances](#)

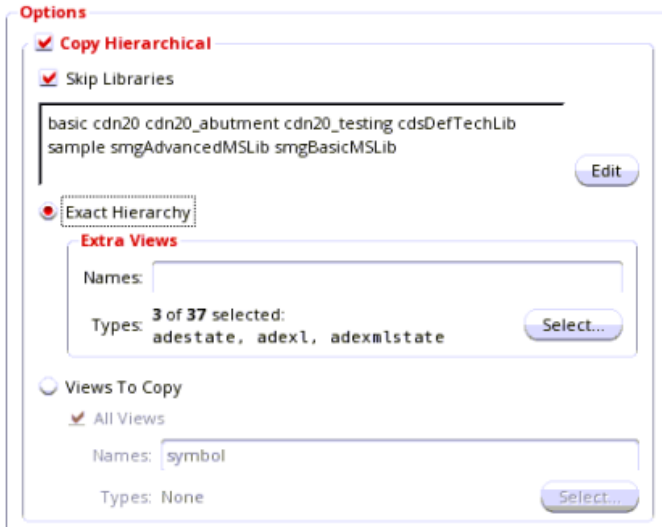
[Adding a Copied Cell to a Category](#)

[Skip Libraries Together with Update Instances](#)

## Copying a Cell Hierarchy in the Library Manager

To traverse the design hierarchy and copy all referenced cells in your design to the destination library, follow these steps:

1. In the *Options* group box, select the *Copy Hierarchical* check box.



2. By default, the *Skip Libraries* check box is also selected. When this check box is selected, cells in the named libraries are not copied and continue to reference their original library. You might want to skip libraries such as reference libraries of contacts, vias, and so on.

You can type the names or click the *Edit* button to open the Skip Libraries Editor dialog box as follows:

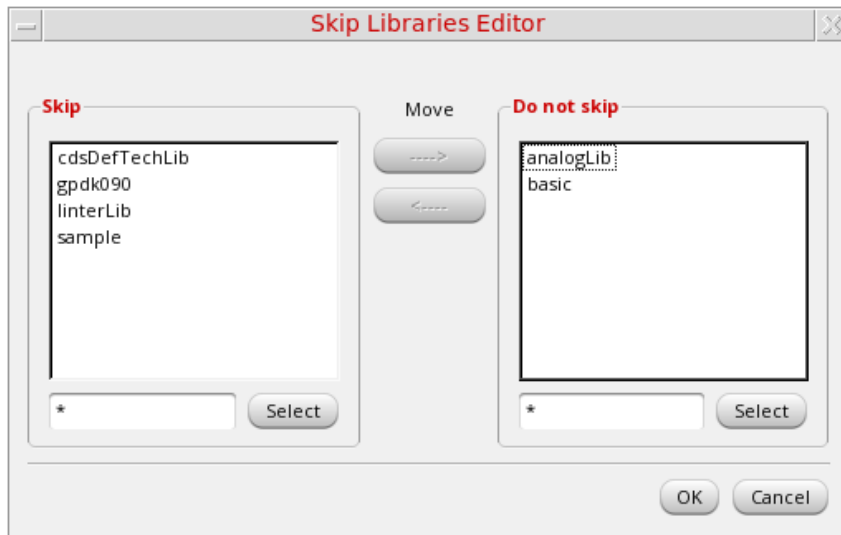
- a. Click library names in the *Skip* list and click the move button in the center to move them to the *Do not skip* list.



## Cadence Library Manager User Guide

### Data Copying

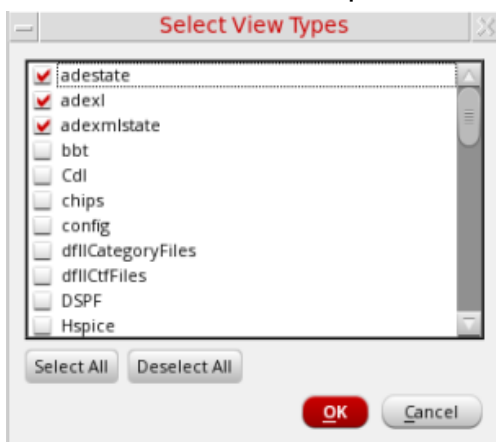
- b. Click *OK*.



3. To limit the search to your design's exact hierarchy when collecting the cellview list for copying, select the *Exact Hierarchy* check box. When this check box is selected, only those cellviews found in the design hierarchy are included in the copy operation.

The *Extra Views* box becomes active. If you include additional view names or view types, the operation includes matching views found in your design hierarchy in the copy operation. If any of these matching views have their own hierarchies, those additional hierarchies are also included.

In the *Names* field, type the names of one or more space-separated views to copy. You can also type a valid filter string (for example, *s\**). Additionally, or optionally, use the *Select* button to select specific view types.



4. By default, the *All Views* check box is selected and the *Views To Copy* option appears deselected. If you leave this check box selected, the copy operation copies all views of

the specified cell. Further, if you copy the cell hierarchically, the copy operation also copies all views of cells instantiated in the specified cell.

To copy a particular set of views, follow these steps:

- a. Deselect the *All Views* check box.

The *Views To Copy* field becomes active.

- b. In the *Names* field, type the names of one or more space-separated views to copy. You can also type a valid filter string (for example, *s\**).

Additionally, or optionally, use the *Select* button to select specific view types.

5. Click *OK*.



#### Video

For an overview on how to reset the changes done on the Copy Cell form, see the [How to make the Library Manager reset changes done on Copy form during the Virtuoso session](#) video.

## Updating Cell Instances

You can use the *Update Instances* check box in the *Options* group box on the Copy Cell form to update the instances in the destination library to use only the copied cells instead of referencing the entire original library and cell names. If you leave the *Update Instances* check box unselected, references to the *From Cell* name are unchanged. For example, all instances of `.../lib/oldCell/symbol` continue to reference the original library and remain `.../lib/oldCell/symbol`.

To update instances in the destination cell to use only the copied cellviews, do the following:

1. In the *Options* group box on the Copy Cell form, select the *Update Instances* check box.

The drop-down list becomes active.

2. From the drop-down list, select one of the following choices:

- ☐ *Of Entire Library*
- ☐ *Of New Copies Only*

3. Click *OK*.

The Library Manager performs the specified copy operation.

## Adding a Copied Cell to a Category

To add a copied cell to a category, do the following:

1. Select the *Add To Category* check box.



The *Add To Category* and *Cells* fields become active.

2. In the *Add To Category* field, type a new or existing category name.
3. In the *Cells* field, type a valid filter string against which to match the cell name that you want to add to the specified category.

By default, \* appears in the *Cells* field, thus matching any copied cell name.

4. Click *OK*.

The copied cell is added to the specified category. If the category does not already exist, it is created.

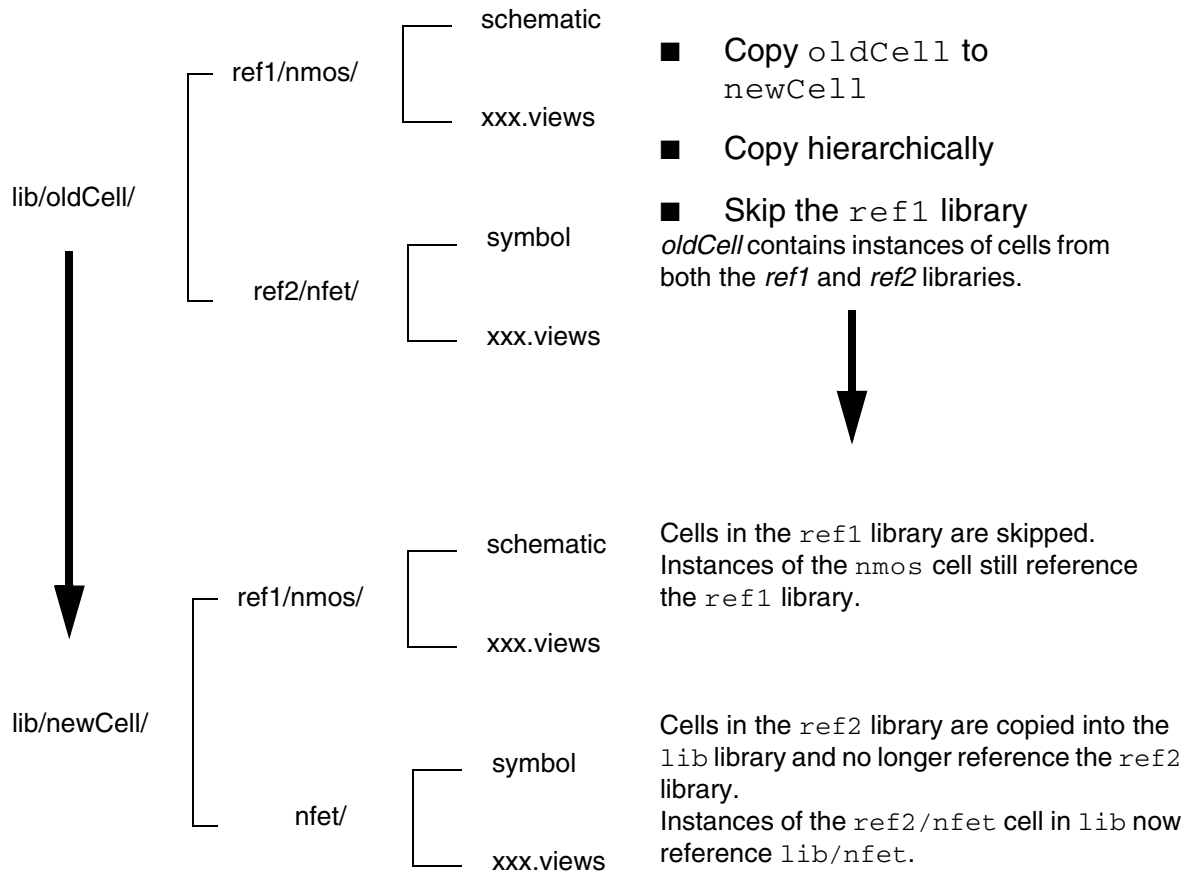
## Skip Libraries Together with Update Instances

## Cadence Library Manager User Guide

### Data Copying

---

The following example shows what happens when you specify a reference library to be skipped in a hierarchical cell copy with the *Update Instances* option enabled.



### ***Related Topics***

[Category Management](#)

## Copying a View in the Library Manager

To copy a view to another name or location, follow these steps:

1. Select a view.
2. Choose *Edit – Copy*.

The Copy View form appears.

The selected library, cell, and view appear in the *Library*, *Cell* and *View* fields in the *From* group box. They also appear in the *Library*, *Cell* and *View* fields in the *To* group box. You can change any or all of these values.

3. In the *To* group box, type the destination library, cell, and view names.
4. To complete the copy view operation, follow the steps for one of the following tasks:

The copy operation does not stop with an error if the zero-sized views are not copied.

## **Copying a View to Another Name or Cell in the Same Library**

Within the same library, you can:

Copy a view to another name for the same cell in the same library:

1. Open the Copy View form.
2. In the *To* group box in the *View* field, type a destination view name.
3. Click *OK*. The *From* view name is copied to the *To* view name for the same cell in the same library.

Copy a view to the same name for a different cell in the same library:

1. Open the Copy View form.
2. In the *To* group box in the *Cell* field, type a destination cell name.
3. Click *OK*. The view is copied to the specified *To* cell name in the same library.

Copy a view to another name for a different cell in the same library:

1. Open the Copy View form.
2. In the *To* group box in the *Cell* field, type a destination cell name.
3. In the *To* group box in the *View* field, type a destination view name.
4. Click *OK*. The *From* view name is copied to the specified *To* cell and view name in the same library.

## **Copying a View to Another Library**

To copy a view to another library, do the following:

1. Open the Copy View form.
2. In the *To* group box in the *Library* field, type or select a destination library name.

You can type a new library name or select an existing library from the drop-down list.

3. In the *To* group box in the *Cell* field, type a destination cell name.

## Cadence Library Manager User Guide

### Data Copying

---

If you do not change the cell name in the *To* group box, the copied cellview displays the same cell name as the original cell.

4. In the *To* group box in the *View* field, type a destination view name.

If you do not change the view name in the *To* group box, the copied view displays the same name as the original view.

5. Click *OK*.

The *From* cellview is copied to the *To* cellview in the *To Library*. If the destination library does not already exist, the New Library form appears, where you can specify a location and design management option for the new library.

### ***Related Topics***

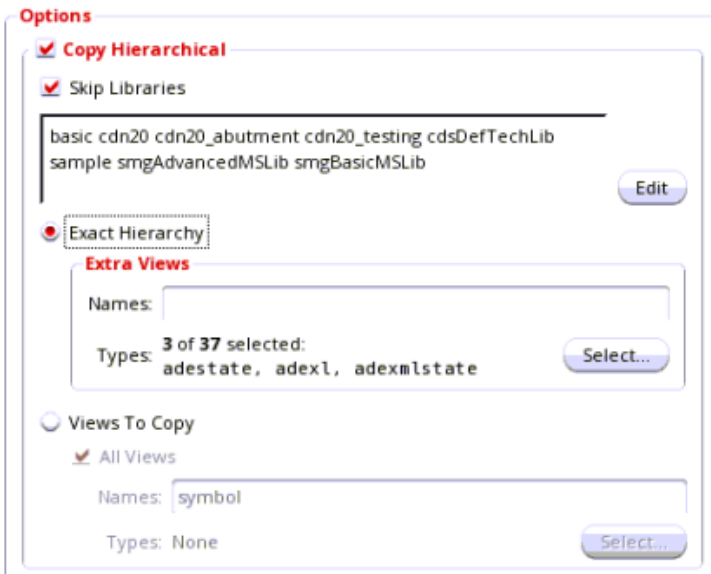
[Copy View Form](#)

[Copying to a New Library](#)

## Copying a View Hierarchy in the Library Manager

To traverse the design hierarchy and copy all views referenced in your design to the destination library, follow these steps:

1. In the *Options* group box, select the *Copy Hierarchical* check box.



2. By default, the *Skip Libraries* check box is also selected. When this check box is selected, cells in the named libraries are not copied and continue to reference their original library. You might want to skip libraries such as reference libraries of contacts, vias, and so on. You can add their names in the text box or click the *Edit* button to open the Skip Libraries Editor dialog box and select the libraries to skip.
3. To limit the search to the exact hierarchy of your design when collecting the cellview list for copying, select the *Exact Hierarchy* check box. When this check box is selected, only those cellviews found in the design hierarchy are included in the copy operation.

The *Extra Views* box becomes active. If you include additional view names or view types in this box, the search is expanded to include any matching views found in your design hierarchy in the copy operation. If any of these matching views have their own hierarchies, those additional hierarchies are also included.

In the *Names* field, type the names of one or more space-separated views to copy. You can also type a valid filter string (for example, *s\**).

Additionally, or optionally, use the *Select* button to select specific view types.

4. The *Views To Copy – All Views* check box appears selected by default. The *Views To Copy* option appears deselected until you deselect the *All Views* check box. If you leave



the check box selected, the copy operation copies all views of the specified cell. Further, if you copy the cell hierarchically, the copy operation also copies all instantiated views.

To copy a particular set of views instead of all views, follow these steps:

- a. Deselect the *All Views* check box.

The *Views To Copy* option becomes active.

- b. In the *Names* field, type the names of one or more space-separated views to copy. You can also type a valid filter string (for example, *s\**).

Additionally, or optionally, use the *Select* button to select specific view types.

All referenced views that match a specified view name or selected view type are copied to the destination.

## Updating View Instances

You can use the *Update Instances* check box in the *Options* group box on the Copy View form to update the instances in the destination library to use only the copied views (instead of referencing the entire original library, cell, and view names). If you leave the *Update Instances* check box unselected, references to the *From View* name are unchanged. For example, all instances of `.../oldLib/oldCell/symbol` continue to reference the original library and remain `.../oldLib/oldCell/symbol`.

To update instances in the destination cell to use only the copied cellviews, do the following:

1. In the *Options* group box on the Copy Cell form, select the *Update Instances* check box.

The drop-down list becomes active.

2. From the drop-down list, select one of the following choices:

- ☐ *Of Entire Library*
- ☐ *Of New Copies Only*

3. Click *OK*.

The Library Manager performs the specified copy operation.

## Related Topics

[Copying a View in the Library Manager](#)

Copying a Library File

## Copying a Library File

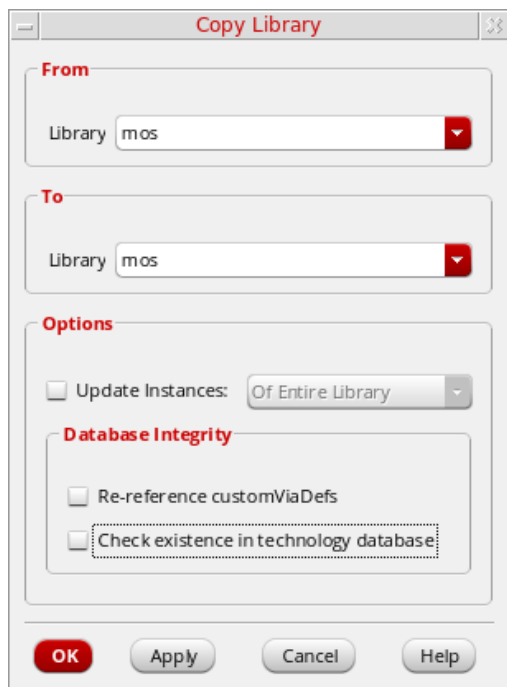
To copy a library file, follow these steps:

1. Select a library file.

If you are viewing library information in *View – Lists* mode, make sure the *Show Files* check box is selected.

2. Choose *Edit – Copy*.

The Copy Library form appears.



The selected library file appears in the *Library* and *File* fields in the *From* group box. It also appears in the *Library* and *File* fields in the *To* group box. You can change any or all of these values.

3. To complete the copy library file operation, follow the steps for one of the following tasks:

## Copying a Library File to Another Name in the Same Library

To copy a library file to another name in the same library, do the following:

1. Open the Copy Library form.
2. In the *To* group box in the *File* field, type a destination file name.

3. Click *OK*.

The *From* file name is copied to the *To* file name in the same library.

## **Copying a Library File to Another Library**

To copy a library file to another library, do the following:

1. Open the Copy Library form.
2. In the *To* group box in the *Library* field, type or select a destination library name.

You can type a new library name or select an existing library from the drop-down list.

3. (Optional) In the *To* group box in the *File* field, type a destination file name.

If you do not change the file name in the *To* group box, the copied file displays the same name as the original file.

4. Click *OK*.

The *From File* is copied to the *To File* in the *To Library*. If the destination library does not already exist, the New Library form appears where you can specify a location and design management option for the new library.

### ***Related Topics***

[Copy Library Form](#)

[Controlling the Display of Library Information](#)

## Copying a Cell File in the Library Manager


To copy a cell file, follow these steps:

1. Select a cell file.

If you are viewing library information in *View – Lists* mode, make sure the *Show Files* check box is selected.

2. Choose *Edit – Copy*.

The Copy Cell File form appears.



The selected cell file appears in the *Library*, *Cell*, and *File* fields in the *From* group box. It also appears in the *Library*, *Cell*, and *File* fields in the *To* group box. You can change any or all of these values.

3. To complete the copy cell file operation, follow the steps for one of the following tasks:

- ☐ Copying a cell file to another name for the same cell in the same library

To copy a cell file to another name for the same cell in the same library,

- a. Open the Copy Cell File form.
- b. In the *To* group box in the *File* field, type a destination cell file name.
- c. Click *OK*. The *From* file name is copied to the *To* file name for the same cell in the same library.

- ☐ Copy a cell file to the same name for a different cell in the same library

To copy a cell file to the same name for a different cell in the same library,

- a. Open the Copy Cell File form.

## Cadence Library Manager User Guide

### Data Copying

---

- b. In the *To* group box in the *Cell* field, type a destination cell name.
- c. Click *OK*. The cell file is copied to the specified *To* cell name in the same library.

- ☐ Copying a cell file to another name for a different cell in the same library

To copy a cell file to another name for a different cell in the same library

- a. Follow the steps from Copying a Cell File in the Library Manager.
- b. In the *To* group box in the *Cell* field, type a destination cell name.
- c. In the *To* group box in the *File* field, type a destination cell file name.
- d. Click *OK*. The *From* file name is copied to the specified *To* cell and file name in the same library.

- ☐ Copying a Cell File to Another Library

To copy a cell file to another library, do the following:

- a. Open the Copy Cell File form.
- b. In the *To* group box in the *Library* field, type or select a destination library name.
- c. You can type a new library name or select an existing library from the drop-down list.
- d. In the *To* group box in the *Cell* field, type a destination cell name.
- e. If you do not change the cell name in the *To* group box, the copied cellview displays the same cell name as the original cell.
- f. In the *To* group box in the *File* field, type a destination cell file name.
- g. If you do not change the cell file name in the *To* group box, the copied file displays the same name as the original file.
- h. Click *OK*.

The *From* cell file is copied to the *To* cell file in the *To Library*. If the destination library does not already exist, the New Library form appears where you can specify a location and design management option for the new library.

### ***Related Topics***

Copy Cell File Form

Controlling the Display of Library Information

## **Cadence Library Manager User Guide**

### **Data Copying**

---

#### Copying to a New Library

## Selecting Text in the Copy Wizard

You can use the Copy Wizard feature to select or deselect items for editing and to select items for copying.

### Selecting Items for Editing in the Copy Wizard

The Copy Wizard table automatically selects multiple cell items that need to be updated whenever a *To Cell* (destination) entry is selected for editing. It selects all the rows that specify the views and files of the associated *From Cell* (source) and changes their *To Cell* values together.

The mandatory cell selection behavior is performed by Library Manager and cannot be disabled.

You can select items for editing in the following ways:

- To select a single library, cell, or view, click the item name.
- To select more than one contiguous item in a row, click and drag the cursor across all the items.
- To select more than one item in a column, do one of the following:
  - ❑ Click and drag the cursor across all the items.
  - ❑ Right-click in the column and choose *Select column* from the pop-up menu.
- To select more than one item not in the same row or column, hold the `Ctrl` key and click each item. Any item not already selected is added to the selection set.

### Deselecting Items in the Copy Wizard

To deselect a selected library, cell, or view, click the item.

To deselect a single item in a group of selected items,

1. Right-click the item.  
A pop-up menu appears.
2. Choose *Deselect*.

**Note:** Clicking one of the selected items deselects all except the item you clicked.



## Cadence Library Manager User Guide

### Data Copying

To deselect an entire selected column, do the following:

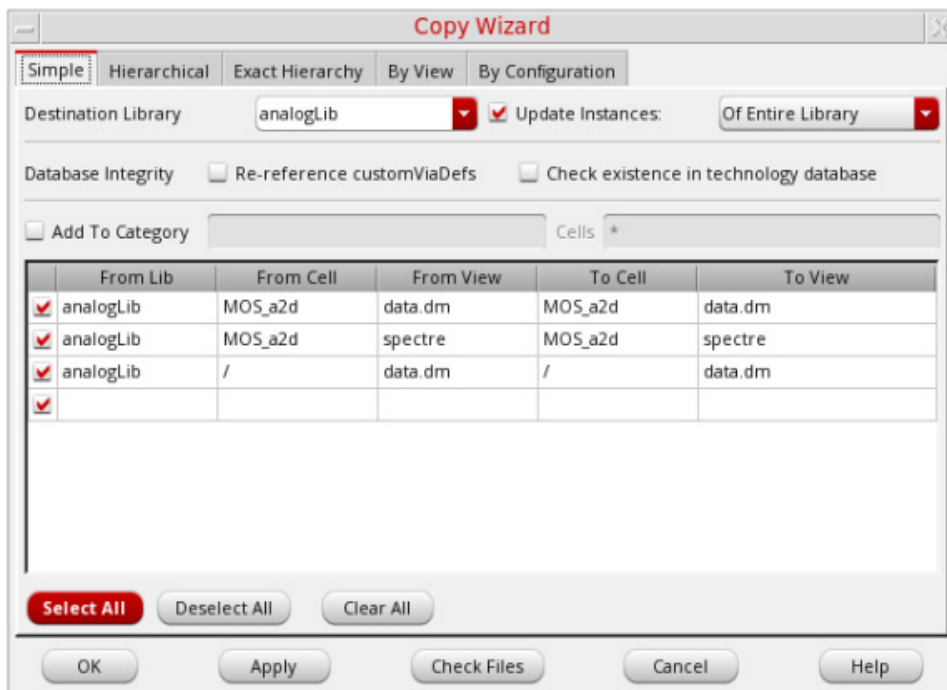
1. Right-click in the column.  
A pop-up menu appears.
2. Choose *Deselect column*.

## Selecting Items for Copying in the Copy Wizard

Each row in the Copy Wizard represents a cellview or a cell file. A selected check box indicates a selected row.

To select or deselect a cellview for copying, do the following:

- Select or deselect the check box to the left of the row.



To select all items for copying, do the following:

- Click *Select All*.

To remove all items from the selection set, do the following:

- Click *Deselect All*.

Items are selected or deselected for copying, not for editing.

***Related Topics***

Selecting Text in the Copy Wizard

## Editing Text in the Copy Wizard

**Note:** You cannot undo changes in the Copy Wizard. If you make a mistake, click *Cancel* to discard changes, then open the Copy Wizard form anew.

You can edit text on the Copy Wizard form in the following ways:

- Editing a Single Item
- Editing All Items in a Selected Set
- Editing All Items in a column

### Editing a Single Item

To change a library, cell, or view name, do the following:

1. Click the item name.

The item is highlighted.

	From Lib	From Cell	From View	To Cell	To View
<input checked="" type="checkbox"/>	analogLib	nmos	auCdl	nmos	auCdl
<input checked="" type="checkbox"/>	analogLib	nmos	auLvs	nmos	auLvs

2. Place the cursor where you want to add or delete text, or drag the cursor across the text to highlight it.
3. Type your changes.
4. Press `Enter`.

### Editing All Items in a Selected Set

To change all items a selected set, do the following:

1. Select all the items you want to change.

	From Lib	From Cell	From View	To Cell	To View
<input checked="" type="checkbox"/>	analogLib	nmos	auCdl	nmos	auCdl
<input checked="" type="checkbox"/>	analogLib	nmos	auLvs	nmos	auLvs
<input checked="" type="checkbox"/>	analogLib	nmos	data.dm	nmos	data.dm
<input checked="" type="checkbox"/>	analogLib	nmos	hspiceD	nmos	hspiceD

**2. Type your changes.**

Only the last item added to the selection set reflects your changes.

**3. Press Enter.**

Alternatively, right-click and choose *Apply Changes* from the pop-up menu.

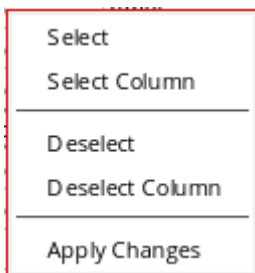
Your edits appear for all selected items.

## Editing All Items in a column

To copy text from one item to all items in the same column, do the following:

**1. Right-click the item you want to copy.**

The pop-up menu appears.



**2. Choose *Select column*.**

All items in the column are selected.

	From Lib	From Cell ▲	From View	To Cell	To View
✓	analogLib	nmos	auCdl	nmos	auCdl
✓	analogLib	nmos	auLvs	nmos	auLvs
✓	analogLib	nmos	data.dm	nmos	data.dm
✓	analogLib	nmos	hspiceD	nmos	hspiceD
✓	analogLib	nmos	spectre	nmos	spectre
✓	analogLib	nmos	symbol	nmos	symbol
✓	analogLib	nmos	symbol_xform	nmos	symbol_xform

**3. Type your changes.**

**4. Right-click in the column to display the pop-up menu.**

**5. Choose *Apply Changes*.**

Your changes are applied to all items in the selected column.

**Note:** While you can make global changes across rows or columns, you are more likely to make global changes across columns.

#### ***Related Topics***

[Selecting Items for Editing in the Copy Wizard](#)

## Performing a Simple Copy Using the Copy Wizard

To perform a simple copy operation, follow these steps:

1. In the Library Manager window, select a library, cell, or view.
2. Choose *Edit – Copy Wizard*.

The Copy Wizard form appears. The default copy option is *Simple*.

**Copy Wizard**

**Simple** Hierarchical Exact Hierarchy By View By Configuration

Destination Library: analogLib Update Instances: ☒ Of Entire Library

Database Integrity: ☐ Re-reference customViaDefs ☐ Check existence in technology database

☐ Add To Category Cells \*

	From Lib	From Cell	From View	To Cell	To View
<input checked="" type="checkbox"/>	analogLib	MOS_a2d	data.dm	MOS_a2d	data.dm
<input checked="" type="checkbox"/>	analogLib	MOS_a2d	spectre	MOS_a2d	spectre
<input checked="" type="checkbox"/>	analogLib	/	data.dm	/	data.dm
<input checked="" type="checkbox"/>					

Select All Deselect All Clear All

OK Apply Check Files Cancel Help

- ☐ If you selected only a library, all cells and views associated with this library appear in the Copy Wizard window.
  - ☐ If you selected a library and a cell, all views associated with this library and cell appear in the Copy Wizard window.
  - ☐ If you selected a library, cell, and view, information for the selected cellview only appears in the Copy Wizard window.
3. Deselect the check box to the left of any item you do not want to include in the copy list.

You can click *Select All* to select all check boxes or *Deselect All* to deselect all check boxes.

## Cadence Library Manager User Guide

### Data Copying

---

4. Change one or more of the following to specify your copy operation:
  - a. In the *Destination Library* field, type or select a new destination library name to copy the selected cellviews to a different library.  
  
If the destination library does not already exist, Library Manager creates it in your working directory during the copy procedure.
  - b. In the *To Cell* list box, edit the contents to copy the selected cellviews to another name.
  - c. In the *To View* list box, edit the contents to copy the selected views to another name.
5. Select the *Update Instances* check box and select the required option from the drop-down list.
6. Select the options in the *Database Integrity* field if you want to update technology data in the destination library after the copy command is completed.
7. Click *Check Files* to preview any copy problems.
  - ☐ Any problems appear on the Copy Problems form.
  - ☐ When there are no problems, the Check is OK prompt appears. Click *OK* to dismiss the prompt and return to the Copy Wizard window.
8. Click *OK*.

The Library Manager performs the specified copy operation.

### ***Related Topics***

[Copy Wizard Form \(Simple Copy\)](#)

[Adding a Copied Cell to a Category](#)

[Editing Text in the Copy Wizard](#)

## Copying a Hierarchy Using the Copy Wizard

To copy a hierarchy of libraries, cells, and views into another library, follow these steps:

1. Select a library, cell, or view.
2. Choose *Edit – Copy Wizard*.

The Copy Wizard form appears.

3. Select the *Hierarchical* tab.

The screenshot shows the 'Copy Wizard' dialog box with the 'Hierarchical' tab selected. The 'Top Library' field contains 'analogLib' and the 'Top Cell' field contains 'MOS\_a2d'. Under 'Views To Copy', the 'Names' field contains 'spectre' and 'Types' is set to 'None'. The 'Skip Libraries' field lists 'basic', 'cdn20', 'cdn20\_abutment', 'cdn20\_testing', 'cdsDefTechLib', 'sample', 'smgAdvancedMSLib', and 'smgBasicMSLib'. The 'Destination Library' is 'analogLib' and 'Update Instances' is checked. A 'Generate Copy List (needed)' button is located below the 'Skip Libraries' field. The 'Database Integrity' section has checkboxes for 'Re-reference customViaDefs' and 'Check existence in technology database'. The 'Add To Category' section has a checkbox and a 'Cells' field with an asterisk. The bottom section is titled 'Complex Copy Operation' and contains instructions: 'Fill in the copy option fields with the desired information and press the 'Generate Copy List' button when finished to obtain a valid list of views to copy.' The footer shows 'CLM-200' and buttons for 'OK', 'Apply', 'Check Files', 'Cancel', and 'Help'.

- ❑ The selected library name appears in the *Top Library* field.
- ❑ If you selected a cell, the cell name appears in the *Top Cell* field.
- ❑ If you selected a view, the view name appears in the *Views To Copy – Names* field.

You can change any or all of these values.

- ❑ If the *Top Cell* field is empty, type a valid cell name.

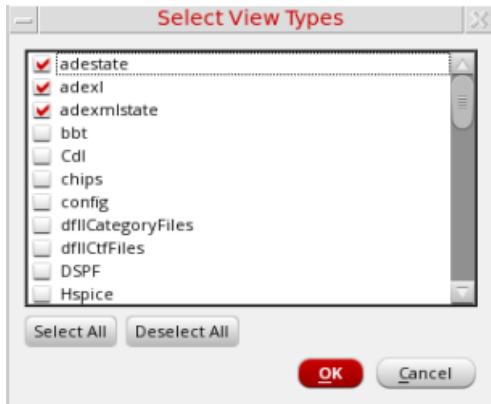


## Cadence Library Manager User Guide

### Data Copying

---

- ❑ If the *Views To Copy – Names* field is empty, type one or more (space-separated) names of views to copy. You can also type a valid filter string. For example, *s\** (to indicate all view names beginning with s) or *\** (to indicate all views). Alternatively, use the *Select* button to select specific view types.



4. In the *Skip Libraries* field, add or remove names of any libraries whose cellviews you want to skip or to copy into the destination library. Alternatively, click the *Edit* button to open the Skip Libraries Editor dialog box and select the libraries to skip.

Cellviews in the libraries named in the *Skip Libraries* field are not copied and continue to reference their original library. (You probably want to skip libraries such as reference libraries of contacts, vias, and so forth.) Use the `skipLibsText` environment variable to include the list of libraries in the *Skip Libraries* field.

5. Click *Generate Copy List*.

The cellviews to copy appear in the copy list. By default, all cellviews are selected for copying.

If the Library Manager finds referenced items whose view names are not specified in the *Views To Copy* field or whose libraries are called out in the *Skip Libraries* field, the View File Summary window appears.

6. In the View File Summary window, click *Yes*.

The list of cellviews not included in the copy list appear on the Referenced Files form.

7. If you want to move one or more of the referenced files to the copy list, do one of the following:

- ❑ In the *Copy* list box, click *No* to change the entry to *Yes* for each referenced file you want to move to the copy list, then click *OK*.
- ❑ Click *Copy All Files* to move all referenced files to the copy list.

The specified cellviews appear in the copy list on the Copy Wizard form.

8. In the *Destination Library* field, type or select a new destination library name to copy the selected cellviews to a different library.
9. Select the *Update Instances* check box and select one the required option from the drop-down list.
10. Select the options in the *Database Integrity* field if you want to update and validate technology data in the destination library after the copy command is completed.
11. Click *OK*.

Library Manager copies the cells in the copy list to the destination library. If the destination library is the same as the top library, Library Manager copies the cells or views to the new names you specified.

If the destination library does not already exist, the New Library form appears, where you can specify a location and design management option for the new library.

### ***Related Topics***

[Copy Wizard Form \(Hierarchical\)](#)

[Adding a Copied Cell to a Category](#)

[Copying to a New Library](#)

[Selecting Text in the Copy Wizard](#)

## Copying an Exact Hierarchy Using the Copy Wizard

To copy an exact hierarchy of libraries, cells, and views into another library (such that only those cellviews found in the design hierarchy are included in the copy operation), follow these steps:

1. Select a library, cell, or view.
2. Choose *Edit – Copy Wizard*.  
The Copy Wizard form appears.
3. Select the *Exact Hierarchy* tab.

The screenshot shows the 'Copy Wizard' dialog box with the 'Exact Hierarchy' tab selected. The dialog has several sections: 'Top Library' (analogLib), 'Top Cell' (MOS\_a2d), and 'Top View' (spectre). Below these is the 'Extra Views' section with a 'Names' field and a 'Types' list showing '3 of 37 selected: adestate, adexl, bbt'. The 'Skip Libraries' section lists 'basic cdn20 cdn20\_abutment cdn20\_testing cdsDefTechLib sample smgAdvancedMSLib smgBasicMSLib'. There is a 'Generate Copy List (needed)' button and a 'Destination Library' dropdown set to 'analogLib'. The 'Update Instances' checkbox is checked, and the 'Update Instances' dropdown is set to 'Of Entire Library'. There are also checkboxes for 'Database Integrity', 'Re-reference customViaDefs', and 'Check existence in technology database'. At the bottom, there is a 'Complex Copy Operation' section with instructions and a 'CLM-200' label. The dialog has 'OK', 'Apply', 'Check Files', 'Cancel', and 'Help' buttons at the bottom.

- ☐ The selected library name appears in the *Top Library* field.
- ☐ If you selected a cell, the cell name appears in the *Top Cell* field.
- ☐ If you selected a view, the view name appears in the *Top View* field.

## Cadence Library Manager User Guide

### Data Copying

---

You can change any or all of these values.

4. If the *Top Cell* field is empty, type a valid cell name.
5. If the *Top View* field is empty, type a valid view name or a list of names separated by spaces.
6. In the *Extra Views – Names*, specify additional space-separated view names or expressions to expand the search to include any matching views found in your design hierarchy in the copy operation. You can also type a valid filter string (for example, *S\**).

Additionally, or optionally, use the *Select* button to select specific view types.

If any of these matching views have their own hierarchies, those additional hierarchies are also included.

7. In the *Skip Libraries* field, add or remove names of any libraries whose cellviews you want to skip or to copy into the destination library. Alternatively, click the *Edit* button to open the Skip Libraries Editor dialog box and select the libraries to skip.

Cellviews in the libraries named in the *Skip Libraries* field are not copied and continue to reference their original library. You might want to skip libraries such as reference libraries of contacts, vias, and so forth. Use the `skipLibsText` environment variable to include the list of libraries in the *Skip Libraries* field.

8. Click *Generate Copy List*.

The cellviews to copy appear in the copy list. By default, all cellviews are selected for copying.

If the Library Manager finds referenced items whose view names are not specified in the *Views To Copy* field or whose libraries are called out in the *Skip Libraries* field, the View File Summary window appears.

9. In the View File Summary window, click *Yes*.

The list of cellviews not included in the copy list appear on the Referenced Files form.

10. On the Referenced Files form, do one of the following if you want to move one or more of the referenced files to the copy list:

- ☐ In the *Copy* list box, click *No* to change the entry to *Yes* for each referenced file you want to move to the copy list, then click *OK*.
- ☐ Click *Copy All Files* to move all referenced files to the copy list.

The specified cellviews appear in the copy list on the Copy Wizard form.

## Cadence Library Manager User Guide

### Data Copying

---

11. In the *Destination Library* field, type or select a new destination library name to copy the selected cellviews to a different library.
12. Select the *Update Instances* check box and select one the required option from the drop-down list.
13. Select the options in the *Database Integrity* field if you want to update and validate technology data in the destination library after the copy command is completed.
14. Click *OK*.

Library Manager copies the cells in the copy list to the destination library. If the destination library is the same as the top library, Library Manager copies the cells or views to the new names you specified.

If the destination library does not already exist, the New Library form appears, where you can specify a location and design management option for the new library.

### ***Related Topics***

[Copy Wizard Form \(Exact Hierarchy\)](#)

[Adding a Copied Cell to a Category](#)

[Copying to a New Library](#)

[Selecting Items for Copying in the Copy Wizard](#)

## Copying Specific View Using the Copy Wizard

To copy only specific views to a different library, follow these steps:

1. Select a library.
2. Choose *Edit – Copy Wizard*.

The Copy Wizard appears.

3. Select the *By View* tab.

The screenshot shows the 'Copy Wizard' dialog box with the 'By View' tab selected. The 'Library' field is set to 'analogLib' and the 'Cell Filter' field is set to 'MOS\_a2d'. In the 'Views To Copy' section, the 'Names' field contains an asterisk '\*' and the 'Types' field indicates '3 of 37 selected: chips, package, smgIpInfo'. A 'Generate Copy List (needed)' button is located below this section. The 'Destination Library' is 'analogLib' and the 'Update Instances' checkbox is checked. There are also checkboxes for 'Database Integrity', 'Re-reference customViaDefs', and 'Check existence in technology database'. At the bottom, there are buttons for 'OK', 'Apply', 'Check Files', 'Cancel', and 'Help'.

- ❑ The selected library name appears in the *Library* field.
- ❑ If you selected a cell, the cell name appears in the *Cell Filter* field.
- ❑ An asterisk appears in the *Names* field in *Views To Copy*.

You can change any or all of these values.

4. In the *Cell Filter* field, type the name of the cell you want to copy.

You can type a specific cell name or a character string to filter patterns in cell names, such as `cc*` or `*a2d`.

5. In the *Views To Copy – Names* field, specify the names of the views you want to copy. You can type a specific view name or a character string to filter patterns in view names, such as `s*`.

Alternatively, or optionally, use the *Select* button to select specific view types.

6. Click *Generate Copy List*.

The cellviews that match the cell and view filter strings appear in the copy list. Cell files are not affected by the *Views To Copy* filter, so all cell files are included in the copy list. By default, all items are selected for copying. In the *Destination Library* field, type or select a new destination library name to copy the selected cellviews to a different library.

7. Select the *Update Instances* check box and select one the required option from the drop-down list.
8. Select the options in the *Database Integrity* field if you want to update technology data in the copied library after the copy command is completed.
9. Click *OK*.

Library Manager copies the cells in the copy list to the destination library. If the destination library is the same as the top library, Library Manager copies the cells or views to the new names you specified.

If the destination library does not already exist, the New Library form appears, where you can specify a location and design management option for the new library.

## ***Related Topics***

[Copy Wizard Form \(By View\)](#)

[Adding a Copied Cell to a Category](#)

[Copying to a New Library](#)

[Selecting Items for Copying in the Copy Wizard](#)

## Copying Specific Cells in a Configuration Using the Copy Wizard

To copy cells in a configuration file to another library:

1. Select the item you want to copy.
2. Choose *Edit – Copy Wizard*.

The Copy Wizard form appears.

3. Select the *By Configuration* tab.

The screenshot shows the 'Copy Wizard' dialog box with the 'By Configuration' tab selected. The 'Library' field contains 'rfTlineLib', the 'Cell' field contains 'mccorn', and the 'Config View' field is empty. The 'Skip Libraries' field contains a list of library names: 'US\_8ths ahdLib analogLib basic cdsDefTechLib functional rfExamples rfLib sample tutorial'. There is an 'Edit' button to the right of the 'Skip Libraries' field. Below the 'Skip Libraries' field is a 'Generate Copy List (needed)' button. The 'Destination Library' field contains 'rfTlineLib' and has a red dropdown arrow. The 'Update Instances' checkbox is checked. The 'Of Entire Library' field has a red dropdown arrow. Below these fields are two checkboxes: 'Database Integrity' and 'Re-reference customViaDefs'. The 'Add To Category' checkbox is unchecked, and the 'Cells' field contains an asterisk (\*).

- ☐ The selected library name appears in the *Library* field.
- ☐ If you selected a cell, the cell name appears in the *Cell* field.
- ☐ If you selected a configuration view, the view name appears on the *Config View* field.

You can change any or all of these values.

4. If the *Cell* field is empty, type a valid cell name.
5. If the *Config View* field is empty, type a valid configuration view name.

In the *Skip Libraries* field, add or remove names of any libraries whose configuration you want to skip or to copy into the destination library. Alternatively, click the *Edit* button to open the Skip Libraries Editor dialog box and select the libraries to skip.



## Cadence Library Manager User Guide

### Data Copying

---

Configuration views in the libraries named in the *Skip Libraries* field are not copied and continue to reference their original library. You might want to skip libraries such as reference libraries of contacts, vias, and so on. Use the `skipLibsText` environment variable to include the list of libraries in the *Skip Libraries* field.

**6.** Click *Generate Copy List*.

The configuration views to copy appear in the copy list. By default, all items are selected for copying.

**7.** In the *Destination Library* field, type or select a new destination library name to copy the selected configuration views to a different library.

**8.** Select the *Update Instances* check box and select one the required option from the drop-down list.

**9.** Select the options in the *Database Integrity* field if you want to update technology data in the destination library after the copy command is completed.

**10.** Click *OK*.

Library Manager copies the cells in the copy list to the destination library. If the destination library is the same as the top library, Library Manager copies the cells or views to the new names you specified.

If the destination library does not already exist, the New Library form appears, where you can specify a location and design management option for the new library.

If the Library Manager cannot open or read the configuration file, check the file permissions on the files making up the configuration or check the configuration using the Hierarchy Editor.

### ***Related Topics***

[Copy Wizard Form \(By Configuration\)](#)

[Adding a Copied Cell to a Category](#)

[Copying to a New Library](#)

[Selecting Items for Copying in the Copy Wizard](#)

[Viewing and Changing File Permissions](#)

## Setting Copy and Rename Preferences

To set preferences for copy and rename operations, do the following:

1. Choose *Edit – Copy Preferences*.

The Copy Preferences form is displayed.

The screenshot shows the 'Copy Preferences' dialog box. At the top, it says 'Settings apply to: Copy, Copy Wizard, Rename, Rename Reference Library'. Below this, there are four sections: 'Library and Cell Property Files', 'Cellview Contents', 'Remote Copy Service', and 'Miscellaneous Settings'. Each section contains radio buttons or checkboxes for different options, with descriptive text for each. At the bottom, there are four buttons: 'Apply', 'Defaults', 'Cancel', and 'Help'.

**Copy Preferences**

Settings apply to: *Copy*  
*Copy Wizard*  
*Rename*  
*Rename Reference Library*

**Library and Cell Property Files**

☒ Automatically add dependent property files to copy sets  
*May increase the number of conflict, overwrite errors*

☐ Do not add dependent property files to copy sets  
*May exclude some files required by the design*

Include properties from: ☒ Libraries ☒ Cells

**Cellview Contents**

☒ Include only the co-managed files of each cellview  
*The smallest complete set defining the design*

☐ Include every file inside each cellview  
*Copies design files plus all additional files*

**Remote Copy Service**

☒ Use session's Copy service when available  
*Enables Copy DFile triggers*

☐ Use Library Manager local Copy engine only  
*Disables Copy DFile triggers*

**Miscellaneous Settings**

☒ Enable file progress monitor

☒ Warn about Rename of managed data (DM)

**Apply** Defaults Cancel Help

## Cadence Library Manager User Guide

### Data Copying

---

2. Choose one of the following *Library and Cell Property Files* settings, which apply to copy operations only:

- ☐ *Automatically add dependent property files to copy sets*

If you choose this setting, you can also select or deselect one or both of the following *Include properties from* check boxes, which are both selected by default:

- ☐ *Libraries*, when selected, indicates that you want to add dependent library property files to a copy set
- ☐ *Cells*, when selected, indicates that you want to add dependent cell property files to a copy set

- ☐ *Do not add dependent property files to copy sets*

3. Choose one of the following *Cellview Contents* settings, which apply to copy operations only:

- ☐ *Include only the co-managed files of each cellview*

- ☐ *Include every file inside each cellview*

4. Choose one of the following *Remote Copy Service* settings, which apply to both copy and rename operations:

- ☐ *Use session's Copy service when available*

- ☐ *Use Library Manager local Copy engine only*

**Note:** For more information about remote copy services, see the description of `cdsLibManager.copyGlobals mpsRadio toggle` in [Using UNIX to Add Settings to a .cdsenv File](#).

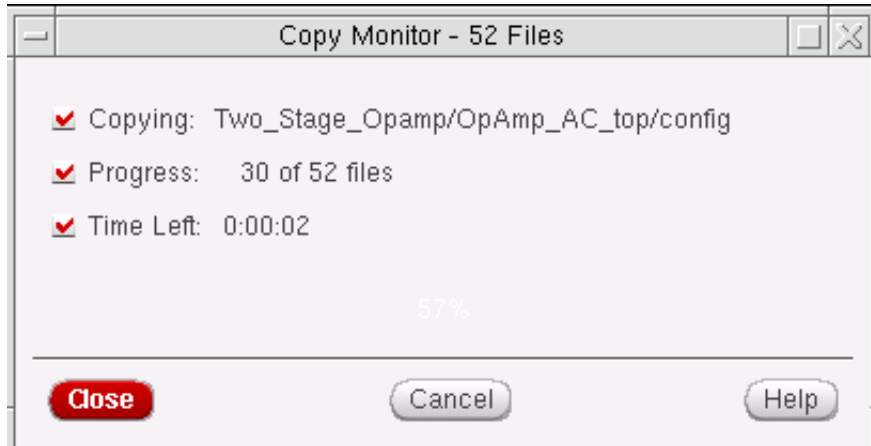
5. Choose *Miscellaneous Settings*:

## Cadence Library Manager User Guide

### Data Copying

---

- ❑ *Enable file progress monitor* causes the Copy Monitor form to appear during the copy operation.



If you click *Cancel*, the Confirm Cancellation form appears.

- ❑ *Warn about Rename of manage data (DM)* causes the Warning: Renaming Managed Data form to appear whenever any part or all of the library you are renaming is under design management (DM).

### ***Related Topics***

[Copy Preferences Form](#)

---

# Design Management

---

Generic Design Management (GDM) is the design management layer that interfaces with the particular design management system you are using. The GDM layer allows applications to interface with different design management systems using a set of basic commands .

When using any design management system that interfaces with GDM, you can optimize the performance of check-in and check-out operations by setting the `DD_GDM_OPTIMIZE` environment variable to `yes` before you start Virtuoso.

## ***Related Topics***

[Setting the Checkin and Checkout Properties](#)

[Canceling Checkout for Properties](#)

[Checking In and Checking Out Categories](#)

[Canceling the Checkout of a Category](#)

[Checking In Files and Properties Automatically](#)

[Auto Checkin Environment Variable Settings](#)

[Submitting Changes to the Design Management System](#)

[Design Management File Status](#)

## Setting the Checkin and Checkout Properties

A property file is treated the same way as any other design file, and the procedure for checking it in is the same as for any other design file.

There are two methods of property and auto check in and check out, each with its own associated forms.

- From the Library Manager, select *Design Manager – Properties – Check In*.
- A second version of property and auto check in and check out, can also however be accessed from within Virtuoso itself, as a result of when object properties are being edited.

Library Manager auto checkin and checkout forms gets displayed for category editing and copy operations.

To check in a property file, follow these steps:

1. Select the item whose property file you want to check in.
2. Choose *Design Manager – Properties – Check In*.

The property file associated with the selected item appears on the Check In form.

3. Click *OK*.

To check out a property file, follow these steps:

1. Select the item whose property file you want to check out.
2. Choose *Design Manager – Properties – Check Out*.

The property file for the selected item appears on the Check Out form.

3. Click *OK*.

## Cadence Library Manager User Guide

### Design Management

The following table summarizes the shell environment variable settings you type in the `.cdsenv` file to control automatic check-in behavior.

<b>CDS_PROMPT_CKIN</b>					
	all	views	file (default)	none	
<b>CDS_AUTO_CKIN</b>	all	Prompt displayed after you close properties, files, or views.	Prompt displayed after you close a view.	Prompt displayed after you close files or properties.	Prompt not displayed.
		Auto check in files, properties, and views.	Auto check in files, properties, and views.	Auto check in files, properties, and views.	Auto check in files, properties, and views.
	views	Prompt displayed after you close properties, files, or views.	Prompt displayed after you close a view.	Prompt displayed after you close files or properties.	Prompt not displayed.
		Auto check in views only.	Auto check in views only.	Auto check in views only.	Auto check in views only.
<b>CDS_AUTO_CKIN</b>	files (default)	Prompt displayed after you close properties, files, or views	Prompt displayed after you close a view.	Prompt displayed after you close files or properties.	Prompt not displayed.
		Auto check in files and properties only.	Auto check in files and properties only.	Auto check in files and properties only.	Auto check in files and properties only.
	none	Prompt displayed after you close properties, files, or views.	Prompt displayed after you close a view.	Prompt displayed after you close properties or files.	Prompt not displayed.
		Never auto check in properties, files, or views.	Never auto check in properties, files, or views.	Never auto check in properties, files, or views.	Never auto check in properties, files, or views.

***Related Topics***

[Check In Form](#)

[Check Out Form](#)

[Canceling Checkout for Properties](#)



## Canceling Checkout for Properties

When you cancel a check-out operation, the software restores your workarea and the project design management repository to the states they were in prior to the check-out. You can cancel a check-out operation if you have not made any changes to a checked-out file or do not want to save any changes you made to a checked-out file.

To cancel a check-out operation, follow these steps:

1. Select the item whose property file check-out operation you want to cancel.
2. Choose *Design Manager – Properties – Cancel Checkout*.

The property file associated with the selected item appears on the Cancel Check Out form.

3. Click *OK*.

The software cancels the check-out operation for each selected file.

### ***Related Topics***

[Cancel Check Out Form](#)

## Checking In and Checking Out Categories

To check in or check out a category, follow these steps:

1. Make sure you can see the category names.
2. Right-click a category name.  
A pop-up menu appears.
3. Choose *Check In* to check in a category, or *Check Out* to check out a category.  
The files associated with the selected category appear on the form.
4. Click *OK*.

### ***Related Topics***

[Check In Form](#)

[Check Out Form](#)

[Controlling the Display of Library Information](#)

[Category Management](#)

## Canceling the Checkout of a Category

You can cancel the check-out of categories to prevent saving the changes to your files.

To cancel a check-out, follow these steps:

1. Right-click a category name.

A pop-up menu appears.

2. Choose *Cancel Check Out*.

The files associated with the selected category appear on the Cancel Check Out form.

3. Click *OK*.

### ***Related Topics***

[Cancel Check Out Form](#)

[Checking In and Checking Out Categories](#)

## Checking In Files and Properties Automatically

By default, when you close properties or files that were automatically checked out, or try to exit a session without closing properties or files that were automatically checked out, the Auto Check In form appears.

To complete the automatic check-in process, do the following:

1. In the drop-down list to the left of the item name, select *yes*.
2. Add the text in the *Comment* text box to specify relevant information about the check-in.

When you add the information, such as version information, about a library or cell check-in, the software attaches a copy of the comment to every cellview in the library.

3. Click *OK*.

To stop the automatic check-in process, do the following:

1. In the drop-down list to the left of the item name, select *no*.
2. Click *OK*.

Alternatively, you can click *Cancel* to stop the automatic check-in process.

To cancel a check-out, do the following:

1. In the drop-down list to the left of the item name, select *cancel checkout*.
2. Click *OK*.

### ***Related Topics***

[Controlling Automatic Checkin Behavior](#)

[Auto Checkin Environment Variable Settings](#)

## Controlling Automatic Checkin Behavior

To control automatic check-in behavior, perform the following steps:

1. On the Auto Checkin form, click *Show Auto Checkin Preferences*.
2. Automatic check-in preferences appear at the bottom of the Auto Checkin form.
3. You can set automatic check-in options separately for cellviews and for properties and files.

To display these preferences from the CIW, do the following:

1. Choose *Options – Checkin Preferences*.
2. The Auto Checkin Preferences form appears.

### Always Ask Me

To force the software to prompt you whenever the automatic check-in process is triggered, do the following:

1. In the drop-down list, select *always ask me*.
2. Click *OK*.

The tool always prompts for automatic check-in. Your changes take effect immediately.

### Never Ask Me

To set the automatic check-in behavior so that you are never prompted, do the following:

1. In the drop-down list, select *never ask me*.

A new drop-down list appears to the right of the original drop-down list.

Whenever you select *never ask me* in either of the *When auto checking in* drop-down lists, you must also select an automatic check-in option.

2. In the drop-down list to the right of *never ask me*, select one of the following choices:

Choice	Behavior
<i>always auto checkin</i>	Always check in the item automatically when closing it
<i>never auto checkin</i>	Do not perform automatic check-in when closing the item

**3. Click *OK*.**

Your changes take effect immediately.

The cellviews are not automatically checked in by default. If you need all the cellviews to be checked-in to the batch mode, then in the *When auto checking in cellViews* section of the Auto Checkin Preferences form, you need to choose the *never ask me* option from the first drop-down list and then choose the *always auto checkin* option from the second drop-down list. Auto checkin of cellview in the batch mode is also possible using the `ddAuto*` SKILL API irrespective of the GUI setting.

***Related Topics***

[Auto Checkin Environment Variable Settings](#)

## Auto Checkin Environment Variable Settings

Set the `CDS_PROMPT_CKIN` and `CDS_AUTO_CKIN` shell environment variables to control the prompting and automatic check-in behavior.

### **CDS\_PROMPT\_CKIN**

`CDS_PROMPT_CKIN` controls 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 Virtuoso Studio design environment product that has both a graphical user interface and automatic check-in capability.

This variable works with `CDS_AUTO_CKIN`.

Valid values are as follows:

`all` specifies that the form appears when either cellview or noncellview data is still checked out.

`none` specifies that the form never appears automatically.

`views` specifies that the form appears only when cellview data is still checked out.

`files` specifies that the form appears only when noncellview data is still checked out. This is default.

### **CDS\_AUTO\_CKIN**

`CDS_AUTO_CKIN` controls whether the software automatically checks in data files when you close properties or files that were automatically checked out or when you exit a Virtuoso session without closing properties or files that were automatically checked out.

- If an application has a graphical user interface, this variable works with `CDS_PROMPT_CKIN`. When `CDS_PROMPT_CKIN` is set to display the Auto Checkin form, the software seeds the form to reflect the value of `CDS_AUTO_CKIN`. When `CDS_PROMPT_CKIN` is not set to display the Auto Checkin form, the software performs the check-in action specified by the value of `CDS_AUTO_CKIN`.
- If the application does not have a graphical user interface, the software ignores `CDS_PROMPT_CKIN` and performs the specified automatic check-in action.

Value values are as follows:

`all` specifies automatic check-in for both cellview and noncellview data.

`none` specifies no automatic check-in operations.

`views` specifies an automatic check-in for cellview data only.

`files` specifies an automatic check-in for noncellview data only. This is default.

### ***Related Topics***

[Checking Out Files and Properties Automatically](#)



## Checking Out Files and Properties Automatically

By default, when you try to open properties, files, or cellviews that you have not checked out, the Auto Checkout form appears.

To complete the auto check-out process, do the following:

1. In the drop-down list to the left of the item name, select *yes*.
2. Click *OK*.

To stop the automatic check-out process, do the following:

1. In the drop-down list to the left of the item name, select *no*.
2. Click *OK*.

Alternatively, you can click *Cancel* to stop the automatic check-out process.

### ***Related Topics***

[Controlling Automatic Checkout Behavior](#)

[Auto Checkout Environment Variable Settings](#)

## Controlling Automatic Checkout Behavior

To control automatic check-out behavior, do the following:

1. On the Auto Checkout form, click *Show Auto Checkout Preferences*.
2. Automatic check-out preferences appear at the bottom of the Auto Checkout form.

To display these preferences from the CIW, do the following:

1. Choose *Options – Checkout Preferences*.
2. The Auto Checkout Preferences form appears.

### Always Ask Me

To force the software to prompt you whenever the automatic check-out process is triggered, do the following:

1. In the drop-down list, select *always ask me*.
2. Click *OK*.

The tool always prompts for automatic check-out. Your changes take effect immediately.

### Never Ask Me

To set the automatic check-out behavior so that you are never prompted, do the following:

1. In the drop-down list, select *never ask me*.

A new drop-down list appears to the right of the original drop-down list.

Whenever you select *never ask me* in either of the *When auto checking out* drop-down lists, you must also select an automatic check-out option.

2. In the drop-down list to the right of *never ask me*, select one of the following choices:

Choice	Behavior
<i>always auto checkout</i>	Always check out the item automatically when opening it
<i>never auto checkout</i>	Do not perform automatic check-out when opening the item

3. Click *OK*.

Your changes take effect immediately.

#### ***Related Topics***

[Auto Checkout Environment Variable Settings](#)

## Auto Checkout Environment Variable Settings

To change the default behavior of the Auto Checkout form, reset the `CDS_PROMPT_CKOUT` and `CDS_AUTO_CKOUT` shell environment variables.

### **CDS\_PROMPT\_CKOUT**

`CDS_PROMPT_CKOUT` controls whether the Auto Checkout form automatically appears when you open data files using a Virtuoso Studio design environment product that has a graphical user interface and automatic check-out capability.

This variable works with `CDS_AUTO_CKOUT`.

Valid values for `CDS_PROMPT_CKOUT` are as follows:

Value	Description
<code>all</code>	(Default) The form appears when you open either cellview or noncellview data
<code>none</code>	The form never appears automatically
<code>views</code>	The form appears only when you open cellview data
<code>files</code>	The form appears only when you open noncellview data

### **CDS\_AUTO\_CKOUT**

`CDS_AUTO_CKOUT` controls whether the software automatically checks out a file when you open it using Virtuoso.

- If an application has a graphical user interface, this variable works with `CDS_PROMPT_CKOUT`. When `CDS_PROMPT_CKOUT` is set to display the Auto Checkout form, the software seeds the form to reflect the value of `CDS_AUTO_CKOUT`. When `CDS_PROMPT_CKOUT` is not set to display the Auto Checkout form, the software performs the check-out action specified by the value of `CDS_AUTO_CKOUT`.
- If the application does not have a graphical user interface, the software ignores `CDS_PROMPT_CKOUT` and performs the check-out action specified by the value of `CDS_AUTO_CKOUT`.

## Cadence Library Manager User Guide

### Design Management

---

Valid values for CDS\_AUTO\_CKOUT are as follows:

Value	Description
all	(Default) Automatic check-out for cellview and noncellview data
none	No automatic check-out operations
views	Automatic check-out for cellview data only
files	Automatic check-out for noncellview data only

### ***Related Topics***

[Controlling Automatic Checkin Behavior](#)

## Submitting Changes to the Design Management System

If you are using a design management system that supports the submit operation, then you can use the *Submit* command to submit items to your project design management repository when you are ready to integrate your design changes into a release as follows:

1. Select a library, cell, view, or file.

2. Choose *Design Manager – Submit*.

If the design management system determines that a submit is needed, the Submit form appears.

3. To remove a file from the selection set for the submit operation, deselect the check box to the left of the file name.

You can remove all files from the selection set by clicking *Deselect All*. You can select all files by clicking *Select All*. For more information about selecting and deselecting items on this form.

4. In the *Comment* field, type text (maximum 100 characters) to describe the design changes associated with this submit operation.

5. In the *Submit Options* group box, select the *Request Name* check box and type a name you want to assign as the Integration Request (IR) name.

If you do not select this box and type a name, the design manager generates a name based on its DM integration behavior, comparable to the `gdmsubmit` Unix command.

6. In the *Submit Options* group box, select the *Use Options* check box and type any submit options specific to your particular design management system you want to use.

7. If you want to use specific options for the IR, turn on *Use Options* and type your options.

8. Click *OK*.

The system submits an IR for your files and notifies you by e-mail (certain DM systems only).

**Note:** This form runs the `gdmsubmit` command.

### ***Related Topics***

[Submit Form](#)

[Selecting Items for Copying in the Copy Wizard](#)

[Generic Design Management \(GDM\) Commands](#)

## Updating an Item Using Design Manager

You can update libraries, cells, views, or files with the latest design data checked in by team members. The *Update* command is active if the selected item can be updated. If an item is checked out, it cannot be updated. The procedure is the same for libraries, cells, views, and files.

To update an item, follow these steps:

1. Select the item you want to update.
2. Choose *Design Manager – Update*.

The appropriate Update form appears.

3. (Optional) In the *Update Options* group box, select the *Update From* check box and type the name of a configuration in the field to update your workarea relative to the specified design management configuration.

For information about the types of configurations you can specify, see the documentation for your design management system.

4. (Optional) In the *Update Options* group box, select the *Use Options* check box and type any update options specific to your particular design management system you want to use.
5. Click *OK*.

The workarea is updated with the latest design data for the selected item.

### ***Related Topics***

[Updating Workarea Using Design Manager](#)

## Updating Workarea Using Design Manager

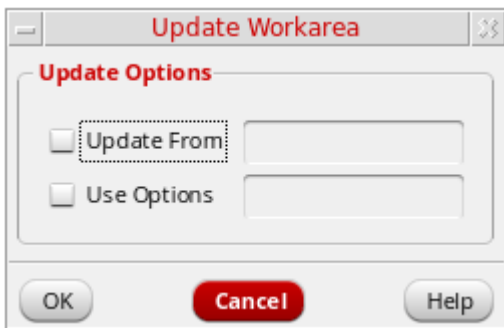
You can update your entire workarea with the latest design data checked in by other users.

**Note:** The *Update Workarea* option can always be run, even when no DM system has been setup, however it has limits.

To update your workarea with the latest design changes, follow these steps:

1. Choose *Design Manager – Update Workarea*.

The Update Workarea form appears.



2. In the *Update Options* group box, select the *Update From* check box and type the name of a configuration in the field to update your workarea relative to the specified design management configuration.

For information about the types of configurations you can specify, see the documentation for your design management system.

3. In the *Update Options* group box, select the *Use Options* check box and type any update options specific to your particular design management system you want to use.
4. Click *OK*.

The software updates all the files in your workarea.

The length of time required to update your workarea depends on the number of cellviews you are updating. Because an update could take a long time, you should choose an appropriate time to run the procedure.

When checking in a library for the first time, add the library to the `project.lib` file (or `cds.lib`) before sharing the data.



***Related Topics***

[Updating an Item Using Design Manager](#)

## Cellview and File Versions

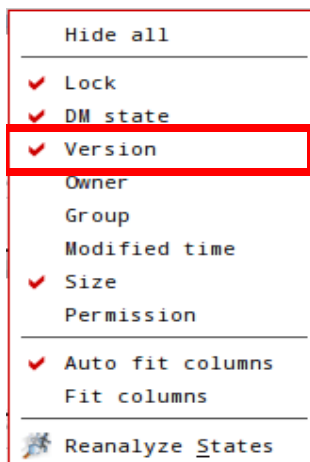
To access the version information for a cellview or file, follow these steps:

1. In the Library Manager, select the cellview or file for which you want to access the version information.
2. Choose *Design Manager – Version Info*.

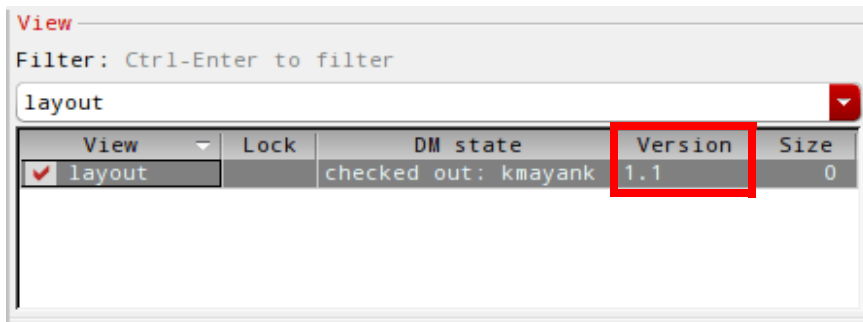
The Version Information form is displayed where the version information is indicated as follows:

- ☐ *Version*: The version of the cellview or file.
- ☐ *Author*: The ID of the user who checked in the version.
- ☐ *Date*: The date when the version was checked in.
- ☐ *Notes*: Any details about the version. The term *Default Version* indicates the most recent version checked in by the user. If the user rolls back the cellview or file to a lower version, it is indicated as *Default Version*.

Alternatively, right-click the title bar of the *View* list box in the Library Manager form and choose *Version* from the pop-up menu.



The *Version* column is added to the *View* list box of the Library Manager form.



## Copying a Version of a Cellview or File

The Copy Cellview Version form is used to copy or export a specific cellview version in the library database, primarily so that it can be viewed side-by-side with another (latest) version of the cellview that is currently being edited in Virtuoso.

You can also access this form directly by selecting *Design Manager – Copy Version*.

To copy a version of a cellview or file:

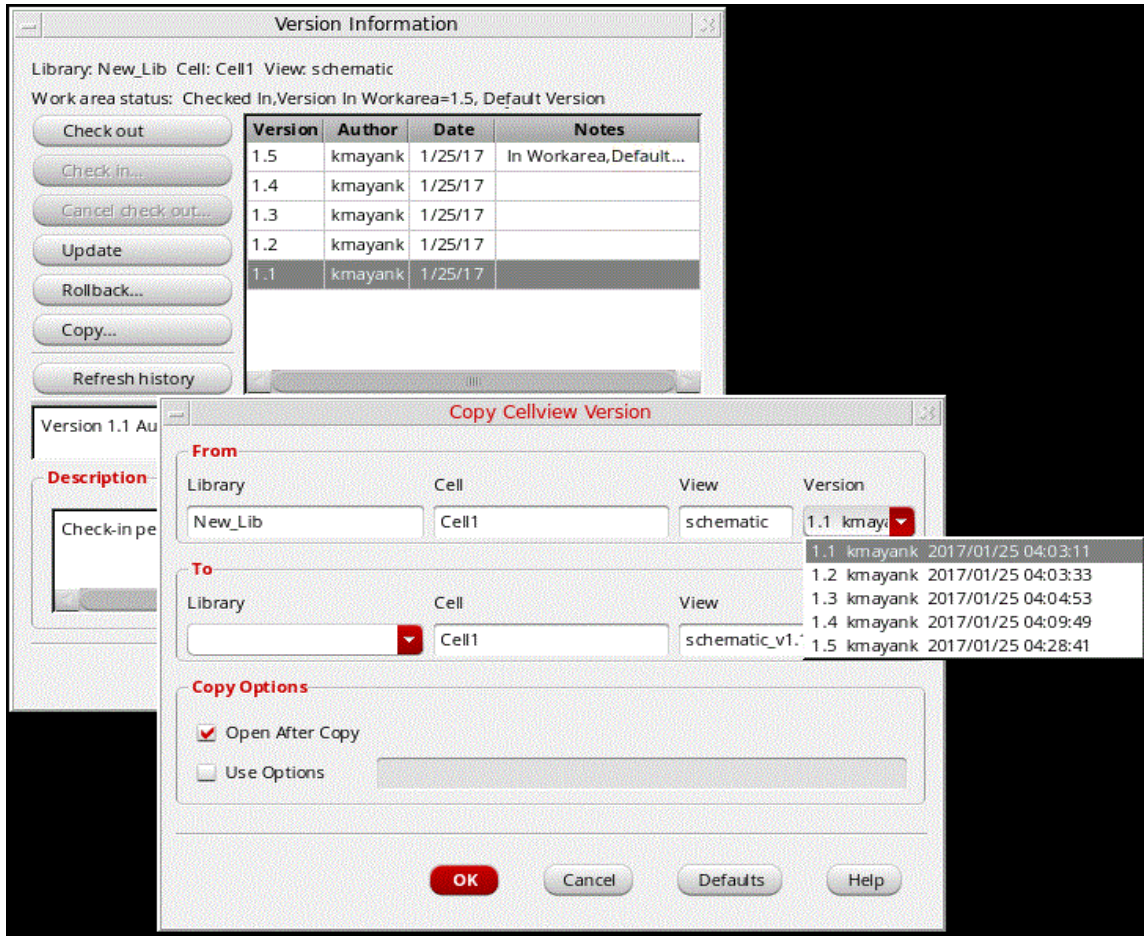
1. Select a cellview that has been checked in.
2. Choose *Design Manager — Version Info*.

The Version Information form appears.

3. Select the version you want to copy.

### 4. Click *Copy*.

The Copy Cellview Version form appears.



The name of the source *Library*, *Cell*, *View*, and *Version* number display in the fields of the *From* group box.

Some default information appears in the fields of the *To* group box. You can change any or all of the *To* values. The default destination view name is the original view name with the version number appended using the format *originalViewName\_v#*. For example, *schematic\_v1.1*.

5. Select the Lib/Cell/View version to be copied from the *Version* pull-down.
6. In the *Library* drop-down field, in the *To* group box, type or select a destination library.
7. (Optional) In the *Copy Options* group box, select the *Open After Copy* check box to open the copied cellview after the copy operation.

8. (Optional) In the *Copy Options* group box, select the *Use Options* check box and type any copy options specific to your particular design management system you want to use.
9. Click *OK*.

The software copies the cellview dependent upon the detailed rules.

### ***Related Topics***

[Version Information Form](#)

[Copy Cellview Version Form](#)

## Cellview Version Rules

The software copies the cellview to the specified destination according to the following guidelines:

- If you type a new library name in the *Library* field, the New Library form appears followed by the Technology File for New Library form.
- If you type an existing cellview name in the *View* field, the Confirm Overwrite form appears.
  - ☐ Click *Yes* to complete the copy operation by overwriting the existing cellview.
  - ☐ Click *No* to cancel the copy operation.

The Destination Already Exists message prompt appears.

Click *OK* and type a different destination on the Copy Cellview Version form.

- If you leave any of the fields blank in the *To* group box, the Missing Destination Name message prompt appears.

To resolve the error, click *OK* and type the missing destination library, cell, or view name.
- If the information you type in the fields of the *To* group box is that same as what appears in the fields of the *From* group box, the Illegal Destination message prompt appears.

To resolve the error, click *OK* and type a valid destination library, cell, and view name.
- If you type an invalid destination name (for example, one containing an illegal character), the Invalid Copy Version Destination Name message prompt appears.

Examples of illegal characters include space, backslash ( \ ), slash ( / ), and punctuation marks such as period ( . ) and comma ( , ).
- If you try to copy a cellview version that is being edited by someone else, the message `Is currently edit locked` appears.
- If you try to copy a cellview version for which you do not have file permission, a message prompt to that effect appears.
- If you try to copy a read-only version of a cellview (or any other file), the Copy Cellview Version Failed message prompt appears.

You cannot overwrite a read-only version of a file. To make the file writable, you must change the access permissions.

***Related Topics***

[Copying to a New Library](#)

## Design Management File Status


Use the DM File Status form to view the design management status of all cells in a library, all views for a cell, and individual files. Additionally, in the DM File Status form, you can view the current status of a design managed library in the *View* list box of the Library Manager. The current DM state gets updated when a cellview is opened or checked out. For design managed libraries, the *View* column also displays applicable status icons as visual updates of the current library, cell, or view state. The tooltip may inform that the current status is that a view has been modified and requires to be checked in.

## Design Management Status Settings

Specify the following status settings for an item:

- *Checked In* specifies that a file is available for check-out to a workarea. The form also shows the project default version number and who created the version.
- *Checked Out* specifies that a file has been checked out to a workarea. The form displays who has checked out the file.
- *Locked* specifies that a file has been locked by you or someone who shares your workarea (sandbox model). The form shows who locked the file and the machine that person is working on. Do not check in a locked file.

## Displaying the Update Needed Icon

The *update needed*  icon is displayed in the Library Manager window in the following scenarios:

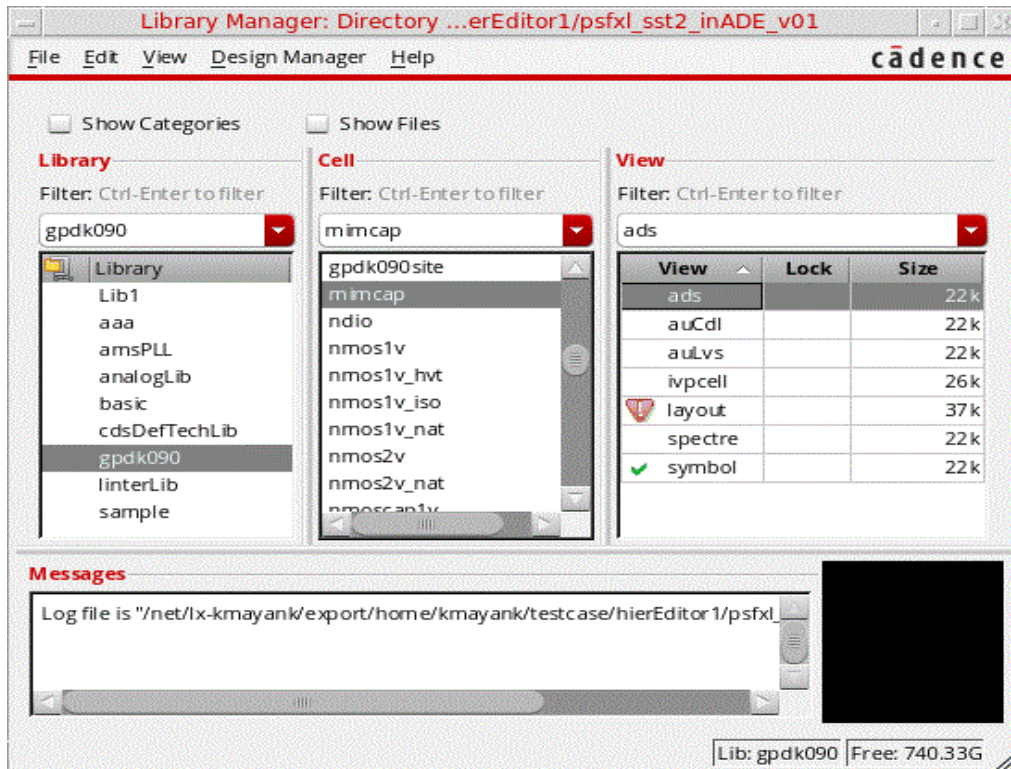
- If the DM status of view files is checked in within your workarea, and an update needs to be applied in the cellview.
- If the DM status of view files is checked in within your workarea but is checked out by another user in another workarea, and an update needs to be applied in the cellview.





## Cadence Library Manager User Guide

### Design Management

In this case, a different icon is displayed if either the update is not needed, or the update needed feature is not available in your DM.



To update the content in the cellview, you need to select the *Design Manager – Update* option from the Library Manager window, or run the `gdmupdate` command from the shell terminal.

After the update is successful, the *updated needed* icon is replaced with either the *checked in*  icon, or the *checked out by others*  icon in the Library Manager window.

For more information on the related GDM command, see [gdmstatus](#).

For more information on the related GDM SKILL function, see [gdmstatus](#).

## Check In, Check Out, and Cancel Check Out Using the DM Status Form

To save you from having to run the separate *Check In*, *Check Out*, and *Cancel Checkout* options in the *Design Manager* menu, these actions can also be performed in the DM Status form.

## Cadence Library Manager User Guide

### Design Management

In the DM Status form, you have the ability to perform these actions for any files shown in the list, which are either from the full list, or from one of the applicable filtered lists.

The screenshot shows the 'DM File Status -> Check In' dialog box. At the top, it says 'Files to Check In [ 1 ]'. Below this are buttons for 'Refresh Status' and 'Library: New\_Lib'. There is a 'Show' button and a dropdown menu currently set to 'All'. A table with two columns, 'Files in library' and 'Status', contains one row: 'Cell1/schematic' with status 'Checked Out Version 1.5 By kmayank'. Below the table are buttons for 'Select All', 'Deselect All', and 'Invert All'. A 'Comment' section with a text area is below the table. At the bottom of the dialog are tabs for 'Manage Set', 'Check In' (which is selected), 'Cancel Check Out', and 'Check Out'. Below the tabs is a 'Check In Options' section with a checkbox for 'Use Options'. At the very bottom are 'OK', 'Cancel', and 'Help' buttons.

	Files in library	Status
1	Cell1/schematic	Checked Out Version 1.5 By kmayank

These check in/out actions, along with the *Refresh Status* command, are also contained in a common tabbed structure in the Check In, Check Out and Cancel Checkout forms. However, when a specific check command is selected, the other command tabs becomes unavailable. For example, if you choose the *Check In* menu option to display the Check In form, then the *Check Out* and *Cancel Check In* tabs contained therein gets disabled. This does not however apply to the Status form which allows you to invoke all of these actions.

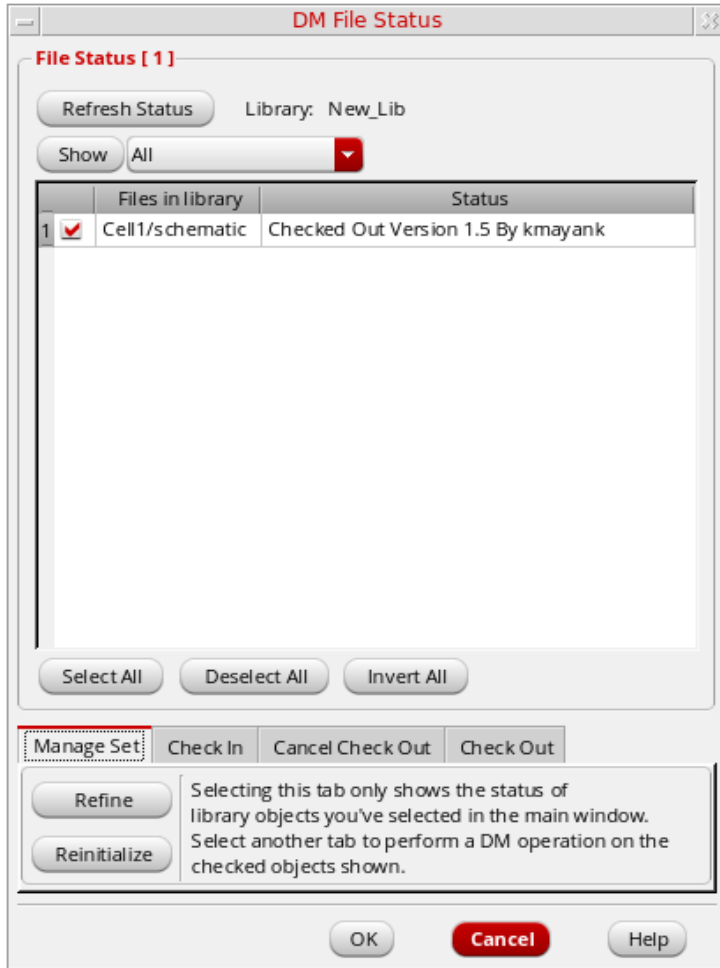
When you perform a check in, check out, or cancel check out from the DM File Status form, a *Substitute DM Command* message gets displayed requesting confirmation of that action.

## Cadence Library Manager User Guide

### Design Management

---

You can however to choose that you do not want this message to be re-displayed in future. In doing so, the `.cdsenv` file gets updated and your preference registered for future actions in this area.



### ***Related Topics***

[DM File Status Form](#)

# **Cadence Library Manager User Guide**

## **Design Management**

---

---

## Library Creation

---

Library Manager helps you in creating new libraries in a design project.

You can create a library using any of the following methods:

- [Creating a New Library in the Library Manager](#)
- [Database Compression Using the oazip Utility](#)
- [Compressing a Library Using Library Manager](#)
- [Compiling an ASCII Technology File](#)
- [Referencing Existing Technology Libraries](#)
- [Attaching a New Library to an Existing Technology Library](#)

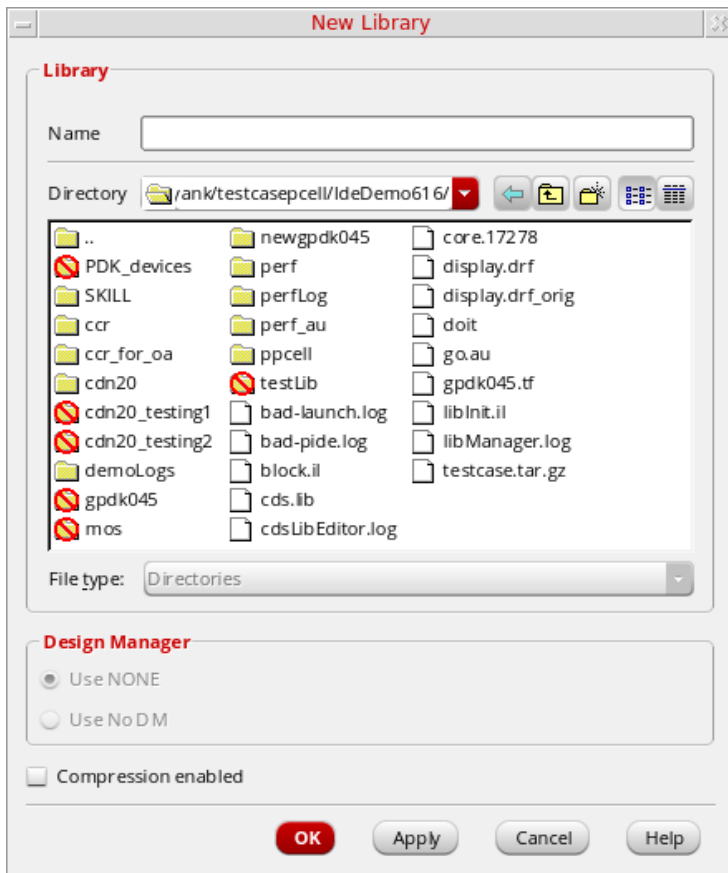
## Creating a New Library in the Library Manager

To create a new library using the Library Manager, follow these steps:

1. In the Library Manager, choose *File – New – Library*. Alternatively, you can click inside the *Library* list box and press **Ctrl+N** on the keyboard.

You can also type the name of the library in the *Library* field of the Library Manager window and press **Ctrl+N** to open the New Library form. In this case, the *Name* field in the New Library form is automatically populated with the name that you have entered in the *Library* field of the Library Manager window.

The New Library Form is displayed.



Creating a new or temporary library within an existing library is not allowed. This is because any directories within a library are handled as cells.

2. In the *Name* field, type the name of the library you want to create.

The new library name cannot be the same as another library.

## Cadence Library Manager User Guide

### Library Creation

---

3. Use the *Directory* navigation tools to specify the destination directory in which you want to create the new library. You can also type a directory path in the *Directory* field. You must have write permission in the directory where you want to create a library.

If you want the library to be under design management control, you must create it in a managed project area. In the *Design Manager* group box, specify whether you want to use a design management system.

- ☐ If you want to use your design management system, select *Use <design management system>* (the default).

When you have the Design Management setup for the new library, the default DM choice can be specified by a cdsenv variable, "ddserv.lib" "dmChoice". If that value matches one of the DM specifications given for the DMTYPE in the cdsinfo.tag file, then the cdsenv value is picked as the default in the New Library form.

The software suggests a design management system if it has been specified with the DMTYPE property in a cdsinfo.tag file that is not library-specific but included in the search path. While creating a new library, the design management (DM) information is stored in the cdsinfo.tag file by default when the DM tool is installed.

- ☐ If you do not want to use design management, select *Use No DM*.

These options gets disabled unless a design management system is available for selection.

4. You can select the *Compression Enabled* check box to write OpenAccess data to this library in a compressed format.
5. Click *OK*.

The Technology File for New Library form is displayed.



6. Choose one of the following technology file options to proceed:

- ☐ *Compile an ASCII technology file*

## Cadence Library Manager User Guide

### Library Creation

---

- ☐ *Reference existing technology libraries*
- ☐ *Attach to an existing technology library*
- ☐ *Do not need process information*

#### ***Related Topics***

[New Library Form](#)

[Virtuoso Software Licensing and Configuration Guide](#)

[The cdsinfo.tag File Location](#)

[Compressing a Library Using Library Manager](#)

[Technology File for New Library Form](#)

[Referencing or Attaching a Technology Library](#)



## Database Compression Using the oazip Utility

OpenAccess supports the ability to save the design databases in a library in a compressed form.

This utility provides the following functionality:

- Processes the OpenAccess databases in a library and compresses them. The compression control value of the library is updated.
- Processes the OpenAccess databases in a library and decompresses the ones that are in compressed form. The compression control value of the library is reset or updated.
- Provides the value of the compression control attribute of a library.
- Scans the OpenAccess databases in a library and reports the databases that do not match the compression control attribute of the library.
- Scans the OpenAccess databases in a library and updates any databases that did not match the compression control attribute of the library.

### Command Syntax

To run `oazip`, enter the following:

```
oazip -lib library {-compress|-decompress|-check|-query|-update} [Optional Arguments]
```

### Arguments

You can use the `-help` or `-h` argument to display command line help. The command line arguments are described in the table below.

Required Argument	
<code>-lib &lt;name&gt;</code>	This required argument specifies the name of the library to process. If this argument is not specified, an error message gets displayed.
One of the Following Arguments is Required	
<code>-compress</code>	If this option is specified, the utility processes the OpenAccess databases in a library, compress the ones in uncompressed form, and reset the compression control value of the library.

## Cadence Library Manager User Guide

### Library Creation

<code>-decompress</code>	If this option is specified, the utility processes the OpenAccess databases in a library, decompress the ones in compressed form, and reset the compression control value of the library.
<code>-query</code>	If this option is specified, the utility reports whether the compression control is specified for the library and what level it is set to.
<code>-check</code>	If this option is specified, the utility reports the OpenAccess databases in the library that are inconsistent with the compression control setting of the library. If there is no compression control specified, the utility lists the databases that are in compressed form. If the compression control is specified, the utility lists those databases that are either in uncompressed form or were written using a compression level different than what the compression control is set to.
<code>-update</code>	If this option is specified, the utility processes the OpenAccess databases in a library and update the ones that are inconsistent with the compression control value of the library.
<b>Optional Arguments</b>	
<code>-h or -help</code>	Display usage information.
<code>-compressLevel &lt;level&gt;</code>	This option specifies the compression level to use for the library. Compression levels refer to the amount of effort the compression algorithm uses to when compressing data. Higher values do not necessarily correspond to better compression efficiency. Compression levels are specified by an integer value between 1 and 9, inclusive. The default value of 1 is suitable for most applications.
<code>-logFile &lt;file&gt;</code>	Specifies the log filename. If this option is omitted, the log filename defaults to <code>oazip.log</code> .
<code>-noInfo &lt;msgIds&gt;</code>	Suppresses the specified INFO messages. <code>msgIds</code> is a quoted, space separated list of numbers. Each number in the list represents the numerical portion of the ID for the message you want to suppress. None of the numbers in the list may be zero. Suppressed messages do not appear on the terminal or in the log file.

## Cadence Library Manager User Guide

### Library Creation

---

<code>-noWarning</code> <code>&lt;msgIds&gt;</code>	Suppresses the specified WARNING messages. <code>msgIds</code> is a quoted, space separated list of numbers. Each number in the list represents the numerical portion of the ID for the message you want to suppress. None of the numbers in the list may be zero. Suppressed messages do not appear on the terminal or in the log file and are not included in the total of WARNING messages displayed in the summary.
<code>-templateFile</code> <code>&lt;file&gt;</code>	<p>Specifies a file containing arguments to oazip. You can specify a template file instead of entering a string of arguments on the command line.</p> <p>If you specify a template file, arguments on the command line have precedence over arguments specified in the file. So, if the same argument exists in the template file and in the command line, the translator uses the argument on the command line.</p> <p>Specify arguments in a template file as follows:</p> <ul style="list-style-type: none"><li>■ Enter arguments in the template file without a dash (-) before the argument.</li><li>■ Enter each argument and value pair on a single line.</li><li>■ Separate the argument from its value using a space or a tab.</li><li>■ Designate comment lines with a # sign as the first character in the line.</li></ul> <p><b>Sample Template File</b></p> <pre># oazip command line arguments: lib                libName logFile            myoazip.log compress</pre>
<code>-v</code>	Prints tool, format, and library version information.
<code>-version</code>	Prints tool and format version information.

### ***Related Topics***

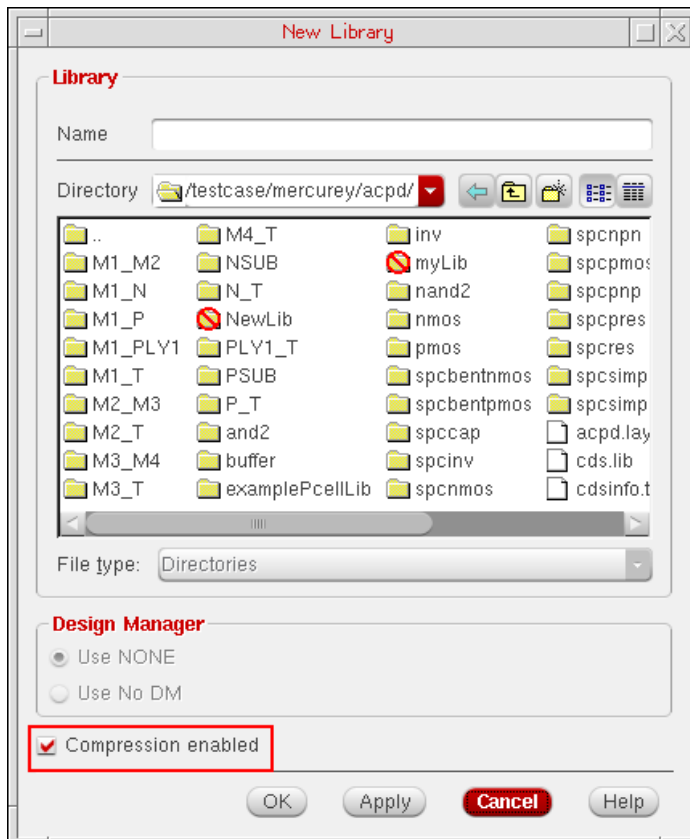
#### Compressing a Library Using Library Manager


## Compressing a Library Using Library Manager

This section describes the procedure of compressing a library.

To compress a library:

1. Select *Compression enabled* to compress the library while creating a library, which results in reducing the disk storage space, offers faster load, and saves transfer time. By default, the *Compression enabled* check box is not selected.

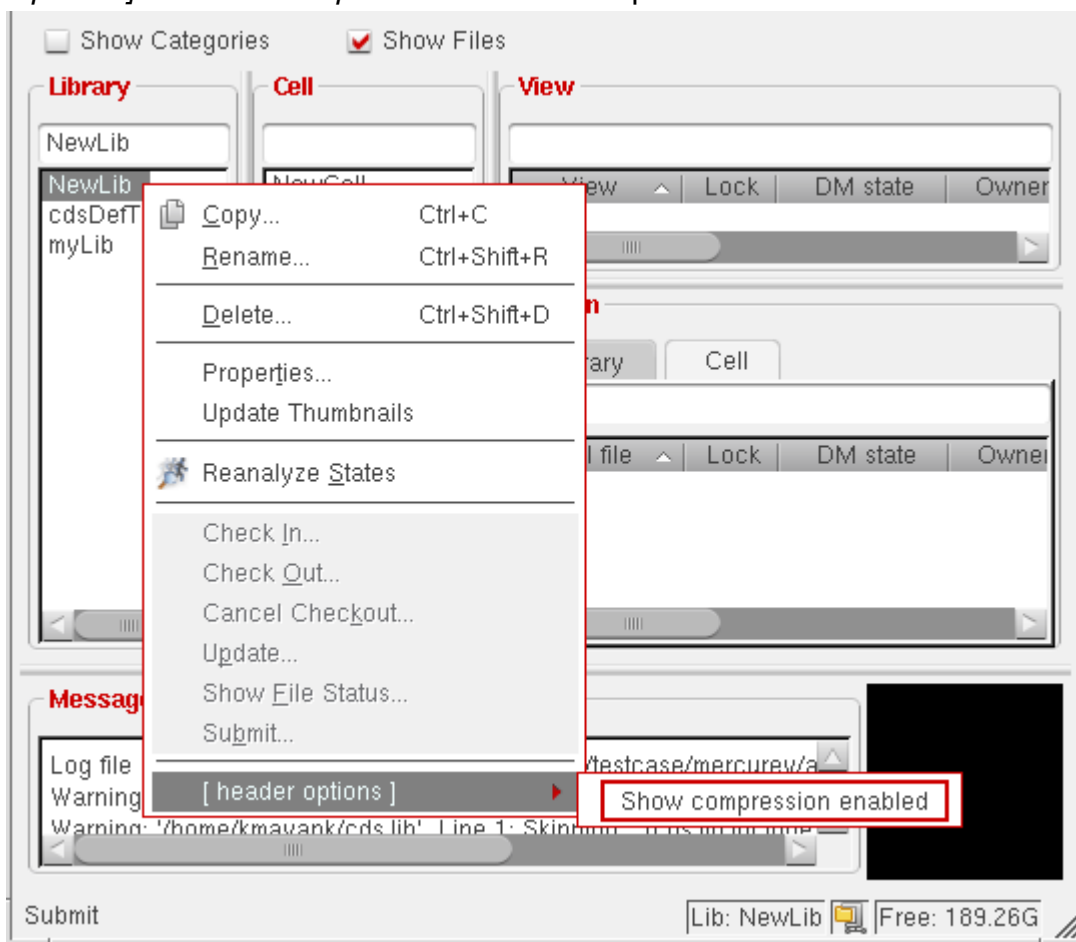


2. Once a new library is created with the *Compression enabled* check box selected, the Library Manager window shows a new compression  icon corresponding to the compressed libraries.

## Cadence Library Manager User Guide

### Library Creation

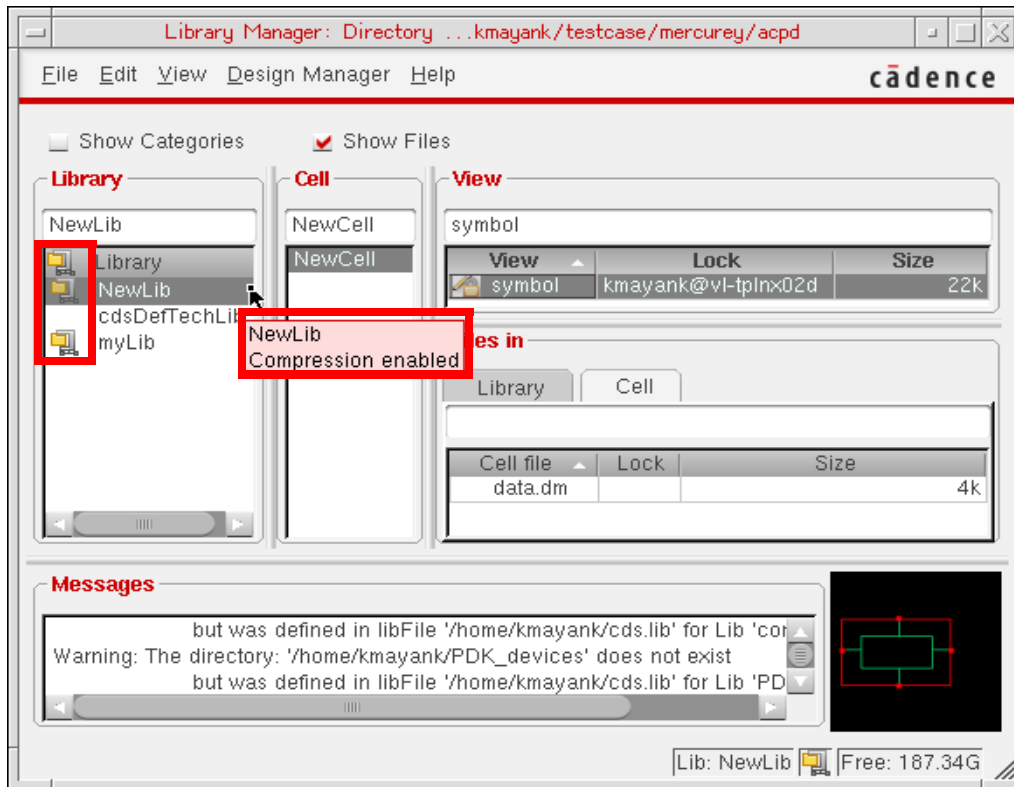
3. To view this icon, right-click the first library in the *Library* list box and select the *[header options] – Show compression enabled* option.



## Cadence Library Manager User Guide

### Library Creation

4. The compression icon is displayed for compressed libraries. Placing the pointer on a compressed library displays a tooltip indicating the compression status as enabled.



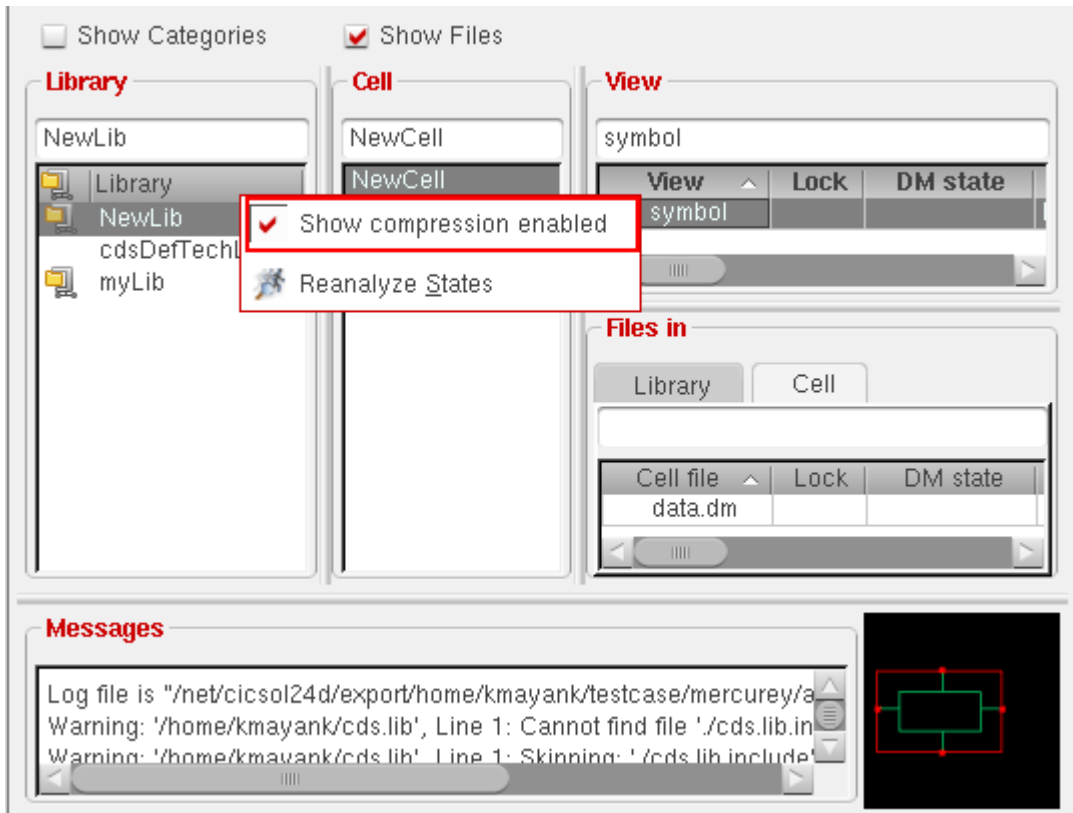
### Important

The `showCompressionDefault` value in `.cdsenv` sets the compression zip icon to be displayed as default if the `$HOME/.libmgr` file is either missing or has no `showCompression` value. For example, `cdsLibManager.showCompression: 1`. Once the value in the `$HOME/.libmgr` file is saved as either 0 or 1, then that value is considered instead of the `.cdsenv` value.

## Cadence Library Manager User Guide

### Library Creation

5. To remove the compression icon from the Library Manager window, you need to right-click the Library header and deselect the *Show compression enabled* option.



Additionally, you can make these enhancements to the Library Manager window:

#### ■ Status Bar Notification

Status bar displays a name of the currently selected library, whether compressed or not, and the approximate free disk space of the volume holding the library at the lower-right corner of the window. For example, values in Megabytes (MB), Gigabytes (GB), or Terabytes (TB) of the free space is displayed.

#### ■ Resizable list boxes

You can also resize all the list boxes together by pressing and holding the `Ctrl` key and dragging the mouse pointer on one of the new list box divider. However, without using the `Ctrl` key pressed, only two adjacent list boxes would be sized at a time.

## ***Related Topics***

### New Library Form

## **Cadence Library Manager User Guide**

### **Library Creation**

---

#### Library Manager Form

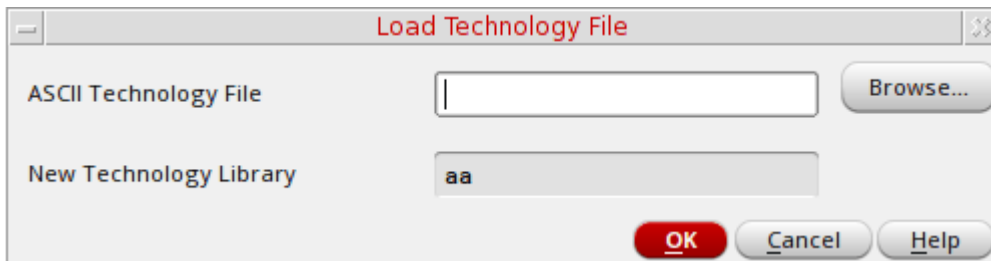


## Compiling an ASCII Technology File

To compile a new technology file and attach it to your new library, follow these steps:

1. From the Technology File for New Library form, select *Compile an ASCII technology file*.
2. Click *OK*.

The Load Technology File form appears.



Your new library name appears in the *New Technology Library* field.

3. In the *ASCII Technology File* field, type the path to a technology file.

You can use your own technology file or one of the following templates from the sample technology files shipped with all Virtuoso applications:

```
install_dir/tools/dfII/samples/default.tf  
install_dir/tools/dfII/samples/mpu.tf
```

where *install\_dir* is the directory in which you installed the Cadence software.

If you want to create a design library with a special technology file but do not know the path to the technology file, type the full hierarchical path to the default technology file `default.tf`. Later, you can use the *Load* command to modify the technology file information. The `default.tf` file acts as a placeholder for your technology file.

4. Click *OK*.

**Note:** If you click *Cancel* instead, the library is created but a technology file is not loaded.

The new design library is created in the specified directory.

### ***Related Topic***

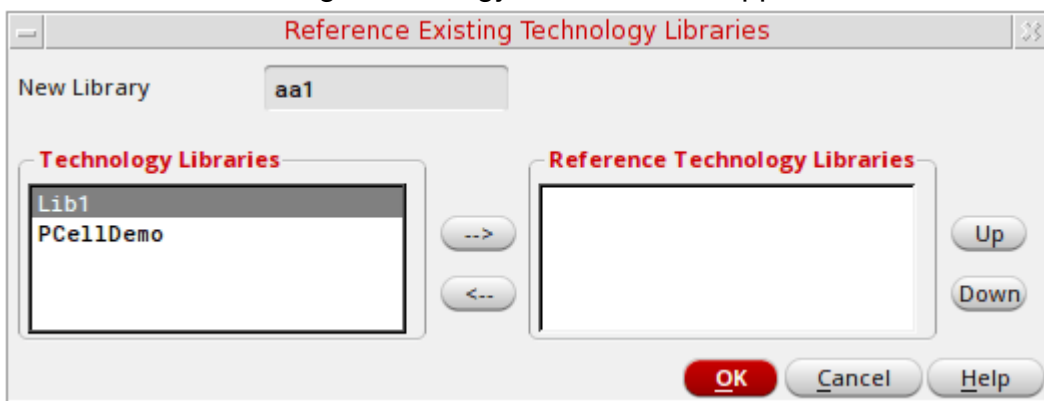
[Load Technology File](#)

## Referencing Existing Technology Libraries

To create a new technology file and reference it to an existing technology library, follow these steps:

1. From the Technology File for New Library form, select *Reference existing technology libraries*.
2. Click *OK*.

The Reference Existing Technology Libraries form appears.



3. Using the arrow buttons (-->, <--), or by double clicking, move the technology libraries that you want your new library to reference from the *Technology Libraries* section to the *Reference Technology Libraries* section.

Only libraries with a local technology library gets listed. If cdsDefTechLib exists, it gets removed as this library is used as the basic, default technology library.

4. You can change the technology library reference priority by selecting a library in the *Reference Technology Libraries* section and choosing to the *Up* or *Down* buttons to change its priority position.

The library at the top of the list has the highest priority when applying rules and constraints to a design.

5. Click *OK*.

Your new library gets created, referencing the selected technology libraries.

### ***Related Topic***

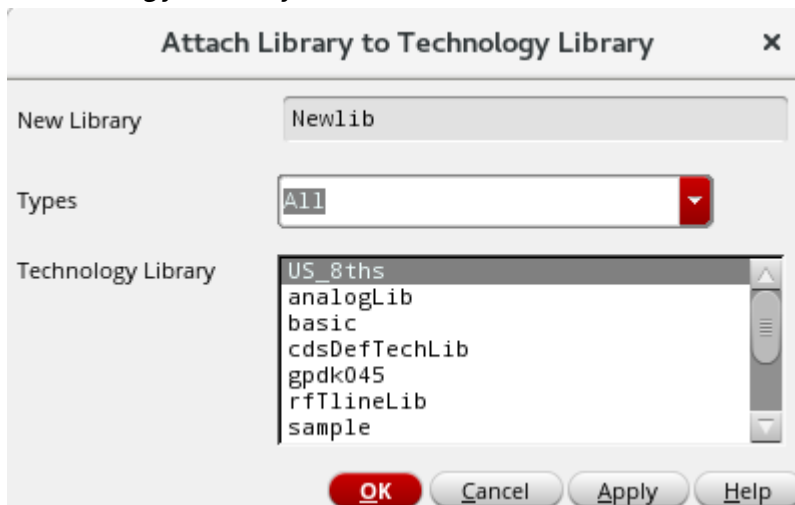
[Attaching a New Library to an Existing Technology Library](#)

## Attaching a New Library to an Existing Technology Library

To attach your new library to a specific technology file, follow these steps:

1. On the Technology File for New Library form, select *Attach to an existing technology library*.
2. Click *OK*.

The Attach Library to Technology Library form appears. The new library name appears in the *New Design Library* field. Technology libraries in your library path appear in the *Technology Library* list box.



3. You can filter the technology library list by selecting the type from the *Types* drop-down.
4. In the *Technology Library* list box, choose the technology library to which you want to attach your new library.

If the `defaultAttachTech` environment variable specifies a valid technology library, it appears selected by default in the list.

5. Click *OK*.

The new library is attached to the specified technology library.

### ***Related Topic***

[defaultAttachTech](#)

## Creating a New Library Without Specifying a Technology File

If you do not plan to design layouts, you do not need a technology file.

To create a new library without specifying a technology file, do the following:

1. On the Technology File for New Library form, select *Do not need process information*.
2. Click *OK*.

Library Manager creates the specified new library.

**Note:** If you are not using a technology file, the system automatically attaches the default technology file `default.tf` when you open a library in Virtuoso.

### ***Related Topics***

[Compiling an ASCII Technology File](#)

[Referencing Existing Technology Libraries](#)

---

## Files in Read-Only Mode

---

When you open a cellview in edit mode or edit the properties of a cellview, the software locks the file to prevent another member of your design team from opening the same file for modification.

When you are working in a design-managed environment or when you have more than one copy of a cellview open, these protective locks sometimes prevent you or other team members from checking in your design or canceling a checkout, even though you have finished your editing and have closed or iconified the cellview.

If you try to check in a locked file then the Cannot Check In message prompt appears.

If you want to continue an ongoing process, such as checking in a library, do the following:

1. On the Cannot Check In message prompt.
2. Click *OK*.

If you want to stop the ongoing process at this point, do the following:

1. On the Cannot Check In message prompt.
2. Click *Cancel*.

Clicking *OK* or *Cancel* does not let you check in the file.

To make it possible for you to check in the locked file, the person who has the file locked must release the lock. To do this, that person can stop editing and close the file or change the file to read-only.

### ***Related Topics***

[Getting a List of Locked Cellviews](#)

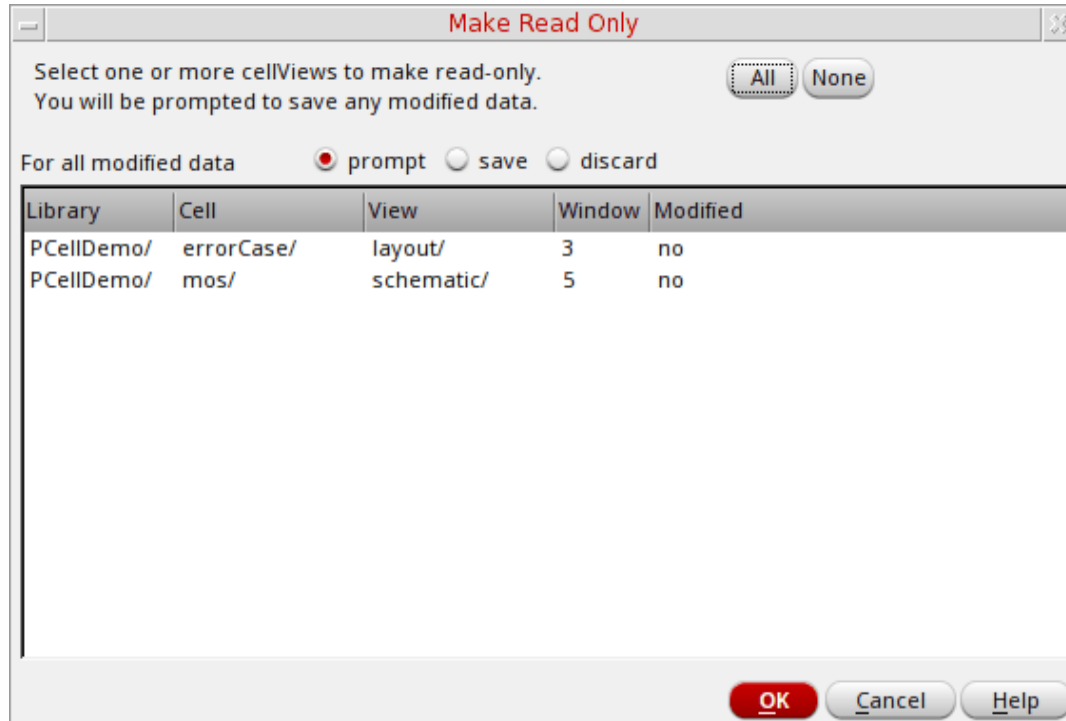
[Making Cellviews Read-Only](#)

## Getting a List of Locked Cellviews

To get a list of locked cellviews:

- In the CIW, choose *File – Make Read Only*.

Any cellviews you have open appear on the Make Read Only form.



### ***Related Topics***

[Making Cellviews Read-Only](#)

## Making Cellviews Read-Only

In order to check in a locked file or open a locked file for editing, the lock on the file must be released. To change files that are open in edit mode to read-only mode so the locks on them are released, follow these steps:

1. Get a list of the cellviews you have locked.
2. On the Make Read Only form, select the cellviews you want to change to read-only.
  - ☐ To select all listed cellviews, click *All*.
  - ☐ To deselect all cellviews, click *None*.
  - ☐ To select a specific cellview, click the line displaying the cellview name.
  - ☐ To select more than one cellview from the list, hold down the `Control` key when you click the line displaying the second and subsequent cellview names.
  - ☐ To deselect cellview names, hold down the `Control` key when you click the selected cellview.

3. Click *OK*.

If you had unsaved changes in the cellview, the Save Cellview message prompt appears.

- ☐ Click *Yes* to save changes
- ☐ Click *No* to discard changes

The Make Read Only form closes and the software changes the selected cellview to be read-only.

This command works only on cellviews, not on property files.

### ***Related Topics***

[Getting a List of Locked Cellviews](#)

[Make Read Only Form](#)

## **Cadence Library Manager User Guide**

### **Files in Read-Only Mode**

---



---

# Category Management

---

To manage a large number of cells in a library, you can assign them to categories. You must have write permission for the library before you can create categories, assign cells to categories, or modify the categories to which the cells belong.

If you have any categories defined in a library, your library directory contains a *libraryName.TopCat* file. The *libraryName.TopCat* file contains a list of all the category files (*categoryName.Cat*) defined in that library. Each *.Cat* file contains a list of cells that are in that category.

## ***Related Topics***

[Creating a Category Using Library Manager](#)

[Editing a Category Using Library Manager](#)

[Deleting a Category Using Library Manager](#)

[Creating a New Category That Includes Subcategories](#)

[Creating a Subcategory in an Existing Category](#)

[Modifying a Category to Include a Subcategory](#)

## Creating a Category Using Library Manager

You can create a new category using the Library Manager and customize it according to your choice.

To create a new category:

**Note:** You must have write permission for the library to create a category.

1. Select a library.
2. Choose *File – New – Category*.

The New Category form is displayed.

The screenshot shows the 'New Category' dialog box. At the top, the title bar says 'New Category'. Below it, there is a text field for 'Category Name' containing 'newCat'. The dialog is divided into two main sections: 'Cells' and 'Sub-Categories'. Each section has two panes: 'Not In Category' and 'In Category'. In the 'Cells' section, the 'Not In Category' pane contains a list of items: 'apc', 'errorCase', 'gscl\_top', and 'mos'. Below this list is a search field with an asterisk and a 'Select' button. The 'In Category' pane is empty, with a similar search field and 'Select' button. Between the panes are two buttons: a right-pointing arrow and a left-pointing arrow. The 'Sub-Categories' section has identical but empty panes and controls. At the bottom of the dialog are four buttons: 'OK' (highlighted with a red border), 'Apply', 'Cancel', and 'Help'.

## Cadence Library Manager User Guide

### Category Management

---

3. In the *Category Name* field, type the name of the new category.

Category names must be unique and legal in the *\_namespace* where you are working.

4. In the *Cells* group box, in the *Not In Category* list box, select the cells you want to include in this category.

You can use *Shift*-click and *Control*-click to modify your selection set.

Use the field to the left of the *Select* button to type a filter string. For example, if you type *buf\** and click *Select*, all cells that start with *buf* are selected.

5. Click the right arrow to move them to the *In Category* list box.

If an error message stating that the software cannot open or cannot write to the category appears, make sure you have write permission for the category files.

6. Click *OK*.

If your library is under design management, the Auto Check In form appears. You can click *OK* to check in the new category.

The New Category form closes and the new category appears in the Library Manager. If this is the first category you are creating in your library, the *Everything* and *Uncategorized* categories are also created.

### ***Related Topics***

New Category Form

Moving Data in List Boxes).

## Editing a Category Using Library Manager

To change the contents of a category:

1. Select the *Category* you want to change.
2. Choose *Edit – Categories – Modify*.

If your library is under design management, the Auto Check Out form appears. You can click *OK* to check out the category.

The Edit Category form appears.

The screenshot shows the 'Edit Category And' dialog box. The 'Category Name' field is set to 'And'. The 'Cells' section has two list boxes: 'Not In Category' (containing aoi21, aoi22, aoi31, aoi32, aoi33, aoi41) and 'In Category' (containing and2, and3, and4, and5, and6). The 'Sub-Categories' section has two list boxes: 'Not In Category' (containing AndOrInvert, Buf, Discretes, FFlops, Inv, Nand) and 'In Category' (empty). The 'Cancel' button is highlighted in red.

The contents of the category you specified appear in the *In Category* list box.

## Cadence Library Manager User Guide

### Category Management

---

3. In the *Cells* group box, do at least one of the following:

- a. In the *Not In Category* list box, select additional cells you want to include in this category and click the right arrow to move them to the *In Category* list box.
- b. In the *In Category* list box, select cells you want to remove from this category and click the left arrow to move them to the *Not In Category* list box.

If an error message stating that the software cannot open or cannot write to the category appears, make sure you have write permission for the category files.

4. Click *OK*.

If your library is under design management, the Auto Check In form appears. You can click *OK* to check in the category.

The Library Manager saves the modifications you made to the category.

### ***Related Topics***

[New Category Form](#)

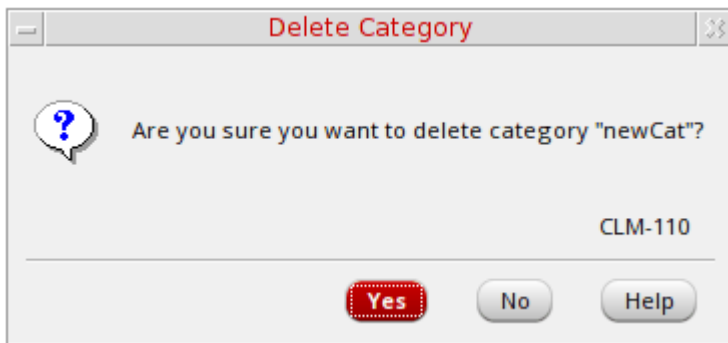
[Moving Data in List Boxes](#)

## Deleting a Category Using Library Manager

To delete a category, follow these steps:

1. Select the category you want to delete.
2. Choose *Edit – Categories – Delete*.

The Delete Category message prompt appears.



3. Click *Yes*.

The Library Manager deletes the category.

### ***Related Topics***

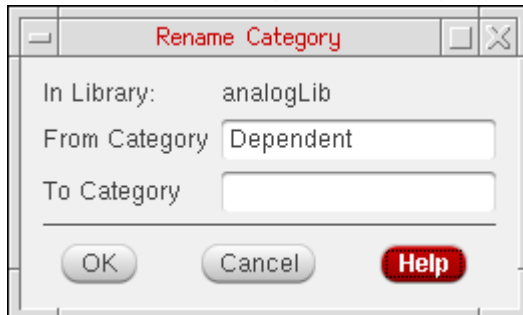
[Editing a Category Using Library Manager](#)

## Renaming a Category Using Library Manager

To rename a category, follow these steps:

1. Select the category you want to rename.
2. Choose *Edit – Categories – Rename*.

The Rename Category message prompt appears.



3. Specify the new category name in the *To Category* text box.
4. Click *OK*.

The Library Manager renames the category.

### ***Related Topics***

[Creating a New Category That Includes Subcategories](#)

## Creating a New Category That Includes Subcategories

To create a new category that includes subcategories, follow these steps:

1. Select a library.
2. Create each of the categories (such as *inputs* and *outputs*) that you want to specify as subcategories by doing the following:

- a. Choose *File – New – Category*.

The New Category form appears.

- b. In the *Category Name* field, type the name of the new category.

You do not have to add any cells to the new categories at this time.

The new categories appear in the Library Manager.

3. After you have created all the categories that are specified as subcategories, create the top-level category in the following way:

- a. Choose *File – New – Category*.

The New Category form appears.

- b. In the *Category Name* field, type the name of the new category.

- c. In the *Sub-Categories* group box, move the categories you want to be subcategories (such as *Inputs* and *Outputs*) to the *In Category* list box.

- d. Click *OK*.

The New Category form closes. The new category has the specified subcategories.

### ***Related Topics***

[New Category Form](#)

[Controlling the Display of Library Information](#)



## Creating a Subcategory in an Existing Category

To create a new subcategory in an existing category, follow these steps:

1. Select the category for which you want to create a subcategory.
2. Choose *Edit – Categories – New Sub-Category*.

The New Sub-Category form appears.

3. In the *Category Name* field, type the name of the new subcategory.
4. In the *Cells* group box, in the *Not In Category* list box, select the cells you want to include in this subcategory.
5. Click the right arrow to move them to the *In Category* list box.

6. Click *OK*.

The New Sub-Category form closes. The Library Manager creates a subcategory in the selected category.

#### ***Related Topics***

[Controlling the Display of Library Information](#)

## Modifying a Category to Include a Subcategory

To modify a category to include another category, follow these steps:

1. Select the category in which you want to include a subcategory.
2. Choose *Edit – Categories – Modify*.

The Edit Category form appears.

3. In the *Sub-Categories* group box, in the *Not In Category* list box, select the category you want included as a subcategory.
4. Click the right arrow to move the selected subcategory to the *In Category* list box.
5. Click *OK*.

The Edit Category form closes. The category you moved is now a subcategory of the modified category.

### ***Related Topics***

[Controlling the Display of Library Information](#)

## **Cadence Library Manager User Guide**

### Category Management

---

---

## Library Manager Customization

---

This section describes the process to customize the menus on the Library Manager form.

- [Library Manager Customizations Using SKILL Functions](#)
- [Customization of Menus Using the cdsLibMgr.il File](#)
- [Callback Function Triggers](#)
- [Library Manager Customizations in Standalone Mode and with Other Processes](#)
- [Restrictions on the Library Manager Customization File](#)
- [Actions in the Library Manager Customization File](#)
- [GUI Objects Supported in the Customization File](#)

## Library Manager Customizations Using SKILL Functions

To customize the Library Manager when you use the Cadence SKILL language, you need the following:

- The `cdsLibMgr.il` file, which contains the following:
  - Callback definition list
  - Callback options and return values
  - Current selection list
- Corresponding SKILL callback functions defined in the Virtuoso Studio design environment

### About `cdsLibMgr.il` File

The extension definition file, `cdsLibMgr.il`, is the starting point for the Library Manager customization. This file defines SKILL extensions specific to the Library Manager and specifies the name of the startup customization file.

The Library Manager `cdsLibMgr.il` file contains the following:

- Callback definition list
- Callback options and return values
- Current selection list

The file is written using the SKILL language. You can use the core Cadence SKILL language as described in the [\*Cadence SKILL Language Reference\*](#). The file can alter existing menus and menu items including removal and change of appearance, as well as add new menus and menu items. You can add menu items to start a SKILL callback in Virtuoso.

A mechanism to start a SKILL callback within the internal Library Manager SKILL interface is not currently provided.

The file is loaded from the first location in the order defined in the Cadence setup search file (`setup.loc`). The typical order used to load this file is:

- the current working directory
- the home directory
- the `install_dir/share` file

## Cadence Library Manager User Guide

### Library Manager Customization

---

When a file is found from one of these locations, the file is loaded and the search stops.

You can customize the name for the `cdsLibMgr.il` file using a default setting in your `.cdsenv` file.

### Callback Definition List

The `lmgrDefineInits()`, `lmgrCreateMenu()`, and `lmgrCreateMenuItem()` functions in the `cdsLibMgr.il` file define callbacks that are started on the Virtuoso Studio design environment program. These functions all use the same syntax for describing the callbacks.

A callback is a list of strings. The first element in the list must be a valid SKILL procedure name in the Virtuoso session. The rest of the list consists of option strings. The following is an example of a callback:

```
' ( "myDeleteObject" "refreshIf" )
```

This callback starts the SKILL procedure `myDeleteObject` in the associated Virtuoso session, which must accept the standard set of arguments described with a single option. The Library Manager redisplay its data if the return value is valid (`t`).

You can define only one list for every callback or map callback function.

The arguments to the callback SKILL procedure include the name of the menu object that this callback is registered with, followed by five arguments representing the current Library Manager selection. The five selection arguments are described in the next section. `Init` and `Close` callbacks defined by `lmgrDefineInits()` do not get passed any arguments.

### Callback Options and Return Values

The option names for a callback function are

- `noOpts` Use this optional placeholder when you want to use no options. You can also use this option to reset preceding options.
- `refresh` Use this option to tell the Library Manager to always regenerate its data display after it runs the callback.
- `refreshIf` Use this option to tell the Library Manager to regenerate its data display after it runs the callback if the return value of the callback indicates success.
- `update` Use this option to tell the Library Manager to always regenerate its own data display and that of Virtuoso after it runs the callback.

## Cadence Library Manager User Guide

### Library Manager Customization

---

- `updateIf` Use this option to tell the Library Manager to regenerate its own data display and that of Virtuoso after executing the callback if the return value of the callback indicates success.

These options can each appear zero or more times. The rightmost options override any to the left, except where noted above. If a sequence ends with `noOpts`, then the entire sequence is interpreted as having no options.

The return value for a correctly run SKILL callback procedure is `t` for success and `nil` if an error occurred.

The Library Manager cannot continue processing if the Library Manager cannot understand the return value of the callback function, such as a database object, a design data (`dd`) object, or an IPC handle.

### Current Selection List

SKILL callback procedures always receive the argument list selection currently specified on the Library Manager form. This list might correspond to a new data object you create, rather than an existing object. The Library Manager sends the data selection specification fully corresponding to the 5.X architecture as a list of five strings:

`LIBNAME CELLNAME VIEWNAME FILENAME CATEGORY.`

If any component of the selection is not specified, it is passed as a string value of `""`.

For example, if nothing at all is selected, then the selection list consists of five empty strings.

Names for a library, cell, and view are located within the designated namespace such as `CDBA`. A file name is always in the file system namespace.

In addition, the current 5.X category in use is sent as the fifth string in the list, which can be a zero length string if categories are disabled in the Library Manager. In general, only commands specific to library categories need to examine the `CATEGORY` parameter. Examples of such commands are `COPY CATEGORY`, `RENAME CATEGORY`, or `CREATE NEW CATEGORY`.

### ***Related Topics***

[Using UNIX to Add Settings to a .cdsenv File](#)

[ImgrCreateMenu](#)

[ImgrCreateMenuItem](#)



## **Cadence Library Manager User Guide**

### **Library Manager Customization**

---

ImgrAddMenuItems

Customization of Menus Using the cdsLibMgr.il File

Restrictions on the Library Manager Customization File

Actions in the Library Manager Customization File

## Customization of Menus Using the `cdsLibMgr.il` File

The Library Manager uses the `cdsLibMgr.il` extension file as follows:

1. The Library Manager loads the `cdsLibMgr.il` extensions file containing menu customization as well as any initialization and termination commands.
2. The modification of GUI menus take place to reflect the customization directives from the extension file.
3. Library Manager then sends the defined initialization commands to the Message Passing Subsystem (MPS) client, which is Virtuoso.
4. The Library Manager GUI interaction triggers MPS callbacks to SKILL routines run in Virtuoso, repeating this sequence as often as necessary.

Custom SKILL routines can modify GUI menu attributes via the MPS interface. These two steps are repeated as often as necessary.

5. The user requests to terminate the Library Manager.
6. The Library Manager sends the defined termination commands to the MPS client.
7. At the end, the Library Manager process terminates.

### ***Related Topics***

[Library Manager Customizations Using SKILL Functions](#)

[Callback Function Triggers](#)

## Callback Function Triggers

You can trigger callbacks at either of two points in the customization process:

- While selecting a menu item
- Before mapping a menu

This feature allows dynamic customization of menus. It is important that you implement pre-map callbacks to be as fast as possible to avoid blocking the X Window System for too long.

### ***Related Topics***

[Library Manager Customizations Using SKILL Functions](#)

[Customization of Menus Using the cdsLibMgr.il File](#)

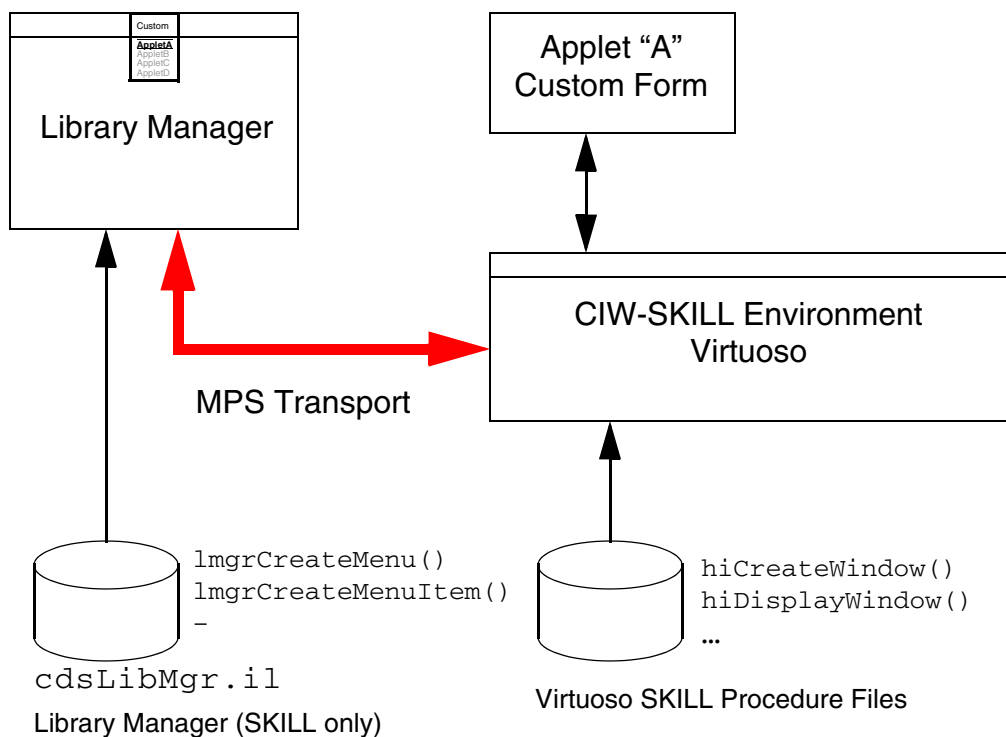
## Library Manager Customizations in Standalone Mode and with Other Processes

### Library Manager Customizations in Standalone Mode

The customization system does not work when you run the Library Manager stand alone. You must run it with a companion Virtuoso Studio design environment process within the same Message Passing Subsystem session. If the customization code is loaded in the Library Manager session that is running in standalone mode, the Library Manager allows only attribute changes to objects such as deletion of unmanaged label attributes. The Library Manager disallows any new objects from becoming active and then issues a warning message.

### Library Manager Customization with Other Processes

The diagram shows how customization is defined and shared among various cooperating processes.



### Related Topics

#### Library Manager Customizations Using SKILL Functions

## Restrictions on the Library Manager Customization File

The following are the restrictions on the Library Manager customization file, `cdsLibMgr.il`:

- The `cdsLibMgr.il` file contains static descriptions of menu customization and is read once at initialization only. Therefore, all possible menus and menu items that you might need during the life of the Library Manager session you must define at startup. Although you cannot define the menus and menu items dynamically at some later time, you can define and leave them unmanaged at startup, so that they can be made visible or invisible dynamically.
- Make sure to Define all objects with string names, which are case sensitive. There are two reserved names with special meaning: `menuBar` and `popup`.
  - `menuBar` refers to the Library Manager top menu bar object, from which all pull-down menus descend.
  - `popup` refers to the pop-up menu selected when you click the middle mouse button over one of the list boxes.
- Define menus as strictly bottom-up, with a strict tree structure. You must define all menu items in a menu before adding them to a menu. To insert menu items into only one pull-down menu. However, a menu item can appear in both a single pull-down and selected pop-up menus. A pull-down menu can appear only once in either the menu bar or in another pull-down menu.
- Use the SKILL API to customize only menus from the menu bar and list pop-ups. No forms are affected by these customizations, although you can partially customize the forms through the `.cdsenv` facilities.
- Pop-up menus cannot contain any submenus. They must have a flat structure.
- You can define only a single callback on pre-map on each menu including the predefined menus, and for all pop-ups. The callback process must be fast.

### ***Related Topics***

[Library Manager Customizations Using SKILL Functions](#)

## Actions in the Library Manager Customization File

The `cdsLibMgr.il` customization file lets you perform the following operations:

- Delete existing predefined menus and menu items from the top menu bar and the pop-up menus.
- Add custom menus to the menu bar or add menu items to predefined menus.



***Changing predefined menus or predefined menu items causes unpredictable behavior.***

- Determine which menu items appear in the pop-up menus for each list box on the Library Manager form (including existing menu items).
- Alter the visual attributes of existing menus and menu items (managed, sensitized, fonts, labels).
- Display anything printed to standard output, such as `printf()` on the Library Manager output pane.

### ***Related Topics***

[Library Manager Customizations Using SKILL Functions](#)

## GUI Objects Supported in the Customization File

The `cdsLibMgr.il` customization file supports the following objects in the graphical user interface:

- Menus, which can contain menu items, toggle items, radio buttons, or separators.
- Menu items as simple buttons. A menu item activates an action callback.
- Toggle buttons. You select a toggle button as a single on/off setting.
- Radio buttons. Radio buttons are mutually exclusive. You select one of several radio buttons displayed.

## Issues with Virtuoso Studio Design Environment SKILL

In order to perform useful custom tasks, any related task functions must be available from SKILL code executed within the Virtuoso process. In some cases, you might need to start an external UNIX process using SKILL IPC (interprocess communication) functions.

When you use a GUI to customize tasks, you must decide whether you want the Library Manager to block the input from the GUI and wait for the task to finish before proceeding. You implement this decision by using the SKILL function `hiCreateAppForm()` with or without the `dontBlock` field set, together with `hiDisplayForm()`.

When you write callback functions, take into account that the Library Manager effectively waits for the return value from the callback in Virtuoso to become available.

## Caution with Pre-Map Callbacks

The Library Manager calls any pre-map callbacks when a menu is to be displayed, whether or not an item is selected, provided there is a map callback function defined for the menu. The Library Manager can stop waiting for this callback to finish after a time-out period has been reached, since the screen becomes locked from all access during the execution of the map callback, where X windows are blocked as well as the Virtuoso session.

The map callback function must be as fast as possible to avoid time-out, which makes your workstation unusable in the interim. You can specify the time-out value in the environment customization file, `.cdsenv`, as a given number of seconds. In the following example, the default value is 5.0 seconds. However, this is much longer than a reasonable response time.

```
cdsLibManager.customize mapTimeout float 5.0
```

## **Cadence Library Manager User Guide**

### Library Manager Customization

---

#### ***Related Topics***

[Cadence User Interface SKILL Reference](#)

[Library Manager Customizations Using SKILL Functions](#)



## Library Manager Customization Using the .cdsenv File

You can use the `.cdsenv` file in the Library Manager as well as in Virtuoso to customize the values and settings on various forms and fields. In addition, some of the settings previously stored in the `.libmgr` file are now saved in the `.cdsenv` file instead.

Library manager obtains form settings from the `.cdsenv` file only and not from the `.cdsinit` file.

The Library Manager saves only the settings that it reads from the `.libmgr` file: the screen location and size of the Library Manager.

You can choose *File – Save Defaults* to save settings in the `.cdsenv` file.

### **.cdsenv File Search Path Order**

The Library Manager searches for the `.cdsenv` file in the following locations, in the specified order:

- `install_dir/tools/dfII/etc/tools/cdsLibManager`  
(This file contains the default settings.)
- `install_dir/tools/dfII/local`
- `$HOME`
- `$CWD`

Virtuoso does not look for the `.cdsenv` file in the current directory by default, although the Library Manager does.

### ***Related Topics***

[Using UNIX to Add Settings to a .cdsenv File](#)

## Using UNIX to Add Settings to a .cdsenv File

To add settings to your `.cdsenv` file, follow the below mentioned steps:

1. Use a text editor to open your `.cdsenv` file.
2. Make the changes you want.
3. Save the file and exit the editor.

You can add Library Manager settings to your `.cdsenv` file using the following format:

```
cdsLibManager.partition varName varType defaultSetting
```

See also `install_dir/tools/dfII/samples/.cdsenv`.

### ***Related Topics***

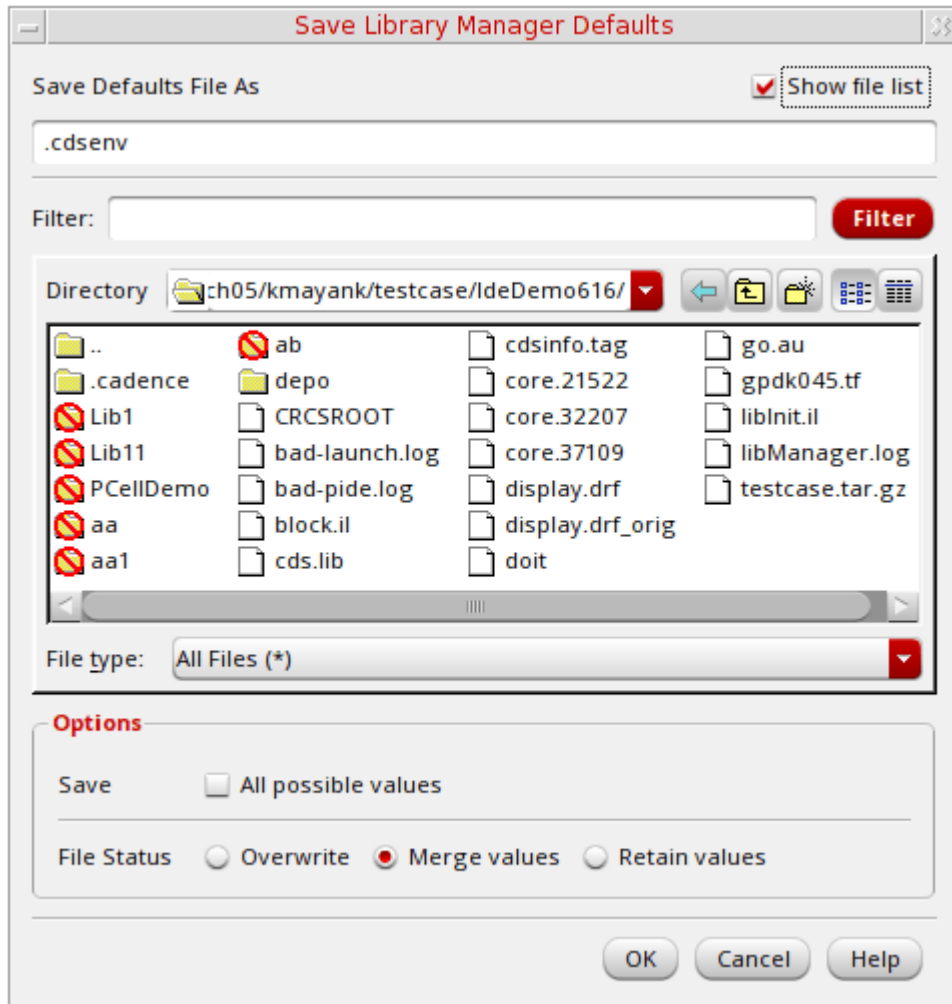
[Library Manager Customization Using the .cdsenv File](#)

## Saving Settings to a .cdsenv File

To save settings to your `.cdsenv` file using the Save Library Manager Defaults form:

1. On the Library Manager form, choose *File – Save Defaults*.

The Save Library Manager Defaults form appears.



2. Use the *Directory* navigation tools to specify the destination directory into which you want to copy the settings file.

You can also type a directory path in the *Save Defaults File As* field.

If you do not specify a directory path, your home directory is used.

3. (Optional) Specify save options:

- ☐ Select the *All possible values* check box to save all possible settings

## Cadence Library Manager User Guide

### Library Manager Customization

---

- ❑ Select an appropriate option from the *File Status* section for the values:
  - *Overwrite Merge values*
  - *Retain values*
- ❑ Click *OK*.

Your settings are saved to the file you specified.

The Save Library Manager Defaults form currently updates only settings that already exist in the `.cdsenv` file.

### ***Related Topics***

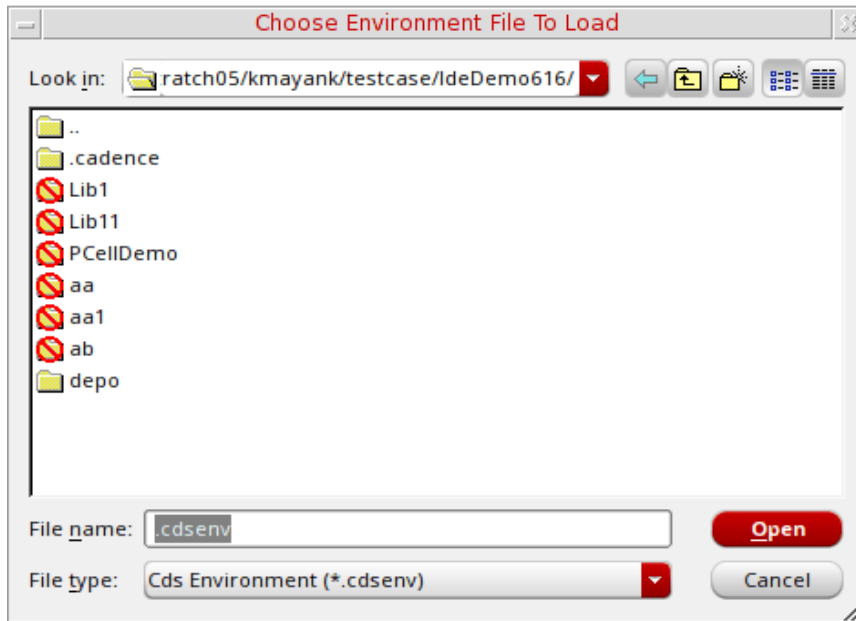
[Save Library Manager Defaults Form](#)

## Loading Settings from a .cdsenv file

To load settings from an environment file (.cdsenv), follow these steps:

1. On the Library Manager form, choose *File – Load Defaults*.

The Choose Environment File To Load form appears.



2. (Optional) Use the navigation tools (list box and toolbar buttons) to specify the source directory from which you want to load the settings file.

You can also type a directory path in the *Look in* field.

If you do not specify a directory path, your home directory is used.

3. In the *File type* field, select one of the following file name filters:

- ☐ *Cds Environment (\*.cdsenv)*
- ☐ *All Files (\*)*

4. In the *File name* field, type the name of the environment settings file you want to load.

Alternatively, you can select the file from the list box above this field.

5. Click *OK*.

The Library Manager loads environment settings from the .cdsenv file you specified.

***Related Topics***

[Choose Environment File To Load Form](#)

## Using the .libsel File to Customize the Library Manager

The `.libsel` file is created by the Library Selector (the libSelect application), also known as the Library Browser. This browser appears when you click the *Browser* button in a Virtuoso form.

The `.libsel` file is created to store the libSelect values so that next time it is invoked it retains settings such as size, screen location, and whether the *Show Categories* check box was selected during the previous session of the Library Manager.

### ***Related Topics***

[Settings in the .libmgr file](#)

## Settings in the .Xdefaults file

You can specify the initial location for the Library Manager by specifying the following resource in your `.Xdefaults` file:

```
cdsLibManager.geometry: widthxheight+xOffset+yOffset
```

For example:

```
cdsLibManager.geometry: 590x580+0+25
```

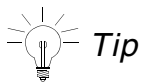
where *width* is the desired width of the window in pixels, *height* is the desired height of the window in pixels, and *xOffset* and *yOffset* specify the distance of the window from the edges of the screen. You can specify *xOffset* and *yOffset* as follows:

<i>+xOffset</i>	The left edge of the window is to be placed <i>xOffset</i> pixels from the left edge of the screen.
<i>-xOffset</i>	The right edge of the window is to be placed <i>xOffset</i> pixels from the right edge of the screen.
<i>+yOffset</i>	The top edge of the window is to be placed <i>yOffset</i> pixels below the top edge of the screen.
<i>-yOffset</i>	The bottom edge of the window is to be placed <i>yOffset</i> pixels above the bottom edge of the screen.

Offsets must be provided as pairs, that is, if you want to specify either *xOffset* or *yOffset*, you must specify both.

To place the window in the corners of the screen, specify the offsets as follows:

+0+0	Places it in the upper left corner
-0+0	Places it in the upper right corner
-0-0	Places it in the lower right corner
+0-0	Places it in the lower left corner



To get the position and size information, you can do the following:

- a. Place the Library Manager window in the location that you want to set as the default.



## Cadence Library Manager User Guide

### Library Manager Customization

---

**b.** In a terminal window, type `xwininfo`.

**c.** Click the Library Manager window.

The terminal window displays information about the Library Manager window, including its position and size. You can use these settings in the `.Xdefaults` file.

### ***Related Topics***

Settings in the `.libmgr` file

## Settings in the .libmgr file

You can also specify a default location and size for the Library Manager in the `.libmgr` file. This file is automatically saved to your home directory each time you exit the Library Manager. The settings in the `.libmgr` file override the settings in the `.Xdefaults` file.

Specify the following settings in the `.libmgr` file:

<code>cdsLibManager.x: 0</code>	Specifies the x-coordinate.
<code>cdsLibManager.y: 82</code>	Specifies the y-coordinate.
<code>cdsLibManager.width: 608</code>	Specifies the width of the window.
<code>cdsLibManager.height: 469</code>	Specifies the height of the window.
<code>cdsLibManager*msgTextWidget.height: 106</code>	Specifies the height of the <i>Messages</i> window.
<code>cdsLibManager.libFilePercent: 34</code>	Specifies the percentage of total list box height used by the <i>Files in Library</i> pane.
<code>cdsLibManager.cellFilePercent: 34</code>	Specifies the percentage of total list box height used by the <i>Files in Cell</i> pane.

### ***Related Topics***

[Settings in the .Xdefaults file](#)

---

## **Library Manager Environment Variables**

---

This topic provides information on the names, descriptions, and graphical user interface equivalents for the Library Manager environment variables.

Only the environment variables documented in this chapter are supported for public use. All other Library Manager environment variables, regardless of their name or prefix, and undocumented aspects of the environment variables described below, are private and are subject to change at any time.

## copyValuesFrom

```
cdsLibManager.addDisplayAttribute copyValuesFrom boolean { t | nil }
```

### Description

Specifies whether the *Copy values from existing library display attribute* option is selected in the Add Library Display Attribute form.

The default is `nil`.

### GUI Equivalent

Command	<i>Display Settings – Library Display Attributes</i>
Form Field	<i>Copy values from existing library display attribute</i>

### Examples

```
envGetVal("cdsLibManager.addDisplayAttribute" "copyValuesFrom")  
envSetVal("cdsLibManager.addDisplayAttribute" "copyValuesFrom" 'boolean t)
```

### ***Related Topics***

[Creating New Library Attributes](#)

## useOptionText

```
cdsLibManager.ckCancel useOptionText string "optionsFile"
```

### Description

Specifies cancel check out options.

The default is " ".

### GUI Equivalent

Command	<i>Design Manager – Cancel Check Out</i>
Form Field	<i>Use Options</i>

### Examples

```
envGetVal("cdsLibManager.ckCancel" "useOptionText")  
envSetVal("cdsLibManager.ckCancel" "useOptionText" 'string "myOptionsFile.txt")
```

### ***Related Topics***

[Cancel Check Out Form](#)

## useOptionsOn

```
cdsLibManager.ckCancel useOptionsOn boolean { t | nil }
```

### Description

Specifies whether checkout cancellation options are enabled.

The default is `nil`.

### GUI Equivalent

Command	<i>Design Manager – Cancel Check Out</i>
Form Field	<i>Use Options</i>

### Examples

```
envGetVal("cdsLibManager.ckCancel" "useOptionsOn")  
envSetVal("cdsLibManager.ckCancel" "useOptionsOn" 'boolean t)
```

### ***Related Topics***

[Cancel Check Out Form](#)

## selectMatches

```
cdsLibManager.ckIn selectMatches toggle (all checkedOut unmanaged checkedIn)
```

### Description

Specifies what items are selected when checking in an item. The choices are:

- `all` means that all items are selected for check-in (the default)
- `ckOut` means that only checked out items are selected for check-in
- `noDM` means that only managed items are selected for check-in
- `ckIn` means that only checked in and writable items are selected for check-in

### GUI Equivalent

Command:     *Design Manager – Check In*

Field:         *Show*

### Examples

The following example returns the current value for the environment variable. The return value of `(t nil nil nil)` indicates that the `all` option is enabled and the other three options are disabled:

```
envGetVal("cdsLibManager.ckIn" "selectMatches")  
(t nil nil nil)
```

The example below enables `ckOut` and `noDM` and disables `all` and `ckInWritable`:

```
envSetVal("cdsLibManager.ckIn" "selectMatches" 'toggle '(nil t t nil))
```

### Related Topics

[Check In Form](#)

## useOptionText

```
cdsLibManager.chkIn useOptionText string "optionsFile"
```

### Description

Specifies check in options. The default is " ".

### GUI Equivalent

Command	<i>Design Manager – Show File Status</i>
Form Field	<i>Check In – Use Options</i>

### Examples

```
envGetVal("cdsLibManager.chkIn" "useOptionText")  
envSetVal("cdsLibManager.chkIn" "useOptionText" 'string "myOptionsFile.txt")
```

### ***Related Topics***

[Check In Form](#)



## useOptionsOn

```
cdsLibManager.ckIn useOptionsOn boolean { t | nil }
```

### Description

Specifies whether check in options are enabled. The default is `nil`.

### GUI Equivalent

Command	<i>Design Manager – Show File Status</i>
Form Field	<i>Check In – Use Options</i>

### Examples

```
envGetVal("cdsLibManager.ckIn" "useOptionsOn")  
envSetVal("cdsLibManager.ckIn" "useOptionsOn" 'boolean t)
```

### ***Related Topics***

[Check In Form](#)

## useOptionText

```
cdsLibManager.ckOut useOptionText string "optionsFile"
```

### Description

Specifies check out options. The default is " ".

### GUI Equivalent

Command	<i>Design Manager – Show File Status</i>
Form Field	<i>Check Out – Use Options</i>

### Examples

```
envGetVal("cdsLibManager.ckOut" "useOptionText")  
envSetVal("cdsLibManager.ckOut" "useOptionText" 'string "myOptionsFile.txt")
```

### ***Related Topics***

[Check Out Form](#)

## useOptionsOn

```
cdsLibManager.ckOut useOptionsOn boolean { t | nil }
```

### Description

Specifies whether check in options are enabled.

The default is `nil`.

### GUI Equivalent

Command	<i>Design Manager – Show File Status</i>
Form Field	<i>Check Out – Use Options</i>

### Examples

```
envGetVal("cdsLibManager.ckOut" "useOptionsOn")  
envSetVal("cdsLibManager.ckOut" "useOptionsOn" 'boolean t)
```

### ***Related Topics***

[Check Out Form](#)

## **addToCategoryName**

```
cdsLibManager.copy addToCategoryName string "category_name"
```

### **Description**

Specifies a category name.

The default is " ".

### **GUI Equivalent**

None

### **Examples**

```
envGetVal("cdsLibManager.copy" "addToCategoryName")  
envSetVal("cdsLibManager.copy" "addToCategoryName" 'string "TitleBlocks")
```

### ***Related Topics***

[Using UNIX to Add Settings to a .cdsenv File](#)

## addToCategoryOn

```
cdsLibManager.copy addToCategoryOn boolean { t | nil }
```

### Description

Specifies whether the add to category option is enabled.

The default is `nil`.

### GUI Equivalent

None

### Examples

```
envGetVal("cdsLibManager.copy" "addToCategoryOn")  
envSetVal("cdsLibManager.copy" "addToCategoryOn" 'boolean t)
```

### *Related Topics*

[Using UNIX to Add Settings to a .cdsenv File](#)

## addToCellsPattern

```
cdsLibManager.copy addToCellsPattern string "filterString"
```

### Description

Specifies a filter string for matching a set of copied cells to add to a category.

The default is `*`.

### GUI Equivalent

None

### Examples

```
envGetVal("cdsLibManager.copy" "addToCellsPattern")  
envSetVal("cdsLibManager.copy" "addToCellsPattern" 'string "sheetSymbol")
```

### ***Related Topics***

[Using UNIX to Add Settings to a .cdsenv File](#)

## **allViewsOn**

```
cdsLibManager.copy allViewsOn boolean { t | nil }
```

### **Description**

Specifies whether to copy all views during a hierarchical copy.

The default is `t`.

### **GUI Equivalent**

None

### **Examples**

```
envGetVal("cdsLibManager.copy" "allViewsOn")  
envSetVal("cdsLibManager.copy" "allViewsOn" 'boolean nil)
```

### ***Related Topics***

[Using UNIX to Add Settings to a .cdsenv File](#)

## **exactHierOn**

```
cdsLibManager.copy exactHierOn boolean { t | nil }
```

### **Description**

Specifies whether to copy the exact hierarchy.

The default is `nil`.

### **GUI Equivalent**

None

### **Examples**

```
envGetVal("cdsLibManager.copy" "exactHierOn")  
envSetVal("cdsLibManager.copy" "exactHierOn" 'boolean t)
```

### ***Related Topics***

[Using UNIX to Add Settings to a .cdsenv File](#)



## existenceCheck

```
cdsLibManager.copy existenceCheck boolean { t | nil }
```

### Description

Specifies whether to select the Check existence in technology database check box in various Copy forms by default.

Default value is `nil`.

### GUI Equivalent

None

### Examples

```
envGetVal("cdsLibManager.copy" "existenceCheck")  
envSetVal("cdsLibManager.copy" "existenceCheck" 'boolean t)
```

### *Related Topics*

[Using UNIX to Add Settings to a .cdsenv File](#)

## extraViews

```
cdsLibManager.copy extraViews string "view_names"
```

### Description

Specifies extra view names to copy.

The default is " ".

### GUI Equivalent

None

### Examples

```
envGetVal("cdsLibManager.copy" "extraViews")  
envSetVal("cdsLibManager.copy" "extraViews" 'string "symbol")
```

### ***Related Topics***

[Using UNIX to Add Settings to a .cdsenv File](#)

## **hierOn**

```
cdsLibManager.copy hierOn boolean { t | nil }
```

### **Description**

Specifies whether to copy hierarchically.

The default is `nil`.

### **GUI Equivalent**

None

### **Examples**

```
envGetVal("cdsLibManager.copy" "hierOn")  
envSetVal("cdsLibManager.copy" "hierOn" 'boolean t)
```

### ***Related Topics***

[Using UNIX to Add Settings to a .cdsenv File](#)

## **rerefCustomVias**

```
cdsLibManager.copy rerefCustomVias boolean { t | nil }
```

### **Description**

Specifies whether to select the Re-reference custom Via Defs check box in various Copy forms by default.

Default value is `nil`.

### **GUI Equivalent**

None

### **Examples**

```
envGetVal("cdsLibManager.copy" "rerefCustomVias")  
envSetVal("cdsLibManager.copy" "rerefCustomVias" 'boolean t)
```

### ***Related Topics***

[Using UNIX to Add Settings to a .cdsenv File](#)

## skipLibsOn

```
cdsLibManager.copy skipLibsOn boolean { t | nil }
```

### Description

Specifies whether to skip libraries during a hierarchical copy.

The default is `t`.

### GUI Equivalent

None

### Examples

```
envGetVal("cdsLibManager.copy" "skipLibsOn")  
envSetVal("cdsLibManager.copy" "skipLibsOn" 'boolean t)
```

### ***Related Topics***

[Using UNIX to Add Settings to a .cdsenv File](#)

## updateChoice

```
cdsLibManager.copy updateChoice toggle (entireLib copiesOnly)
```

### Description

Specifies what part of the library needs to be updated when updating instances.

- `entireLib` means to update the entire library when updating instances (the default)
- `copiesOnly` means to update the new copies only when updating instances

### GUI Equivalent

Command:     *Edit – Copy*

Field:         *Update Instances*

### Examples

The following example returns the current value for the environment variable. The return value of `(t nil)` indicates that the `entireLib` option is enabled and `copiesOnly` is disabled:

```
envGetVal("cdsLibManager.copy" "updateChoice")  
(t nil)
```

The example below enables `copiesOnly` and disables `entireLib`:

```
envSetVal("cdsLibManager.copy" "updateChoice" 'toggle '(nil t))
```

### ***Related Topics***

[Copy Library Form](#)

[Copy View Form](#)

[Using UNIX to Add Settings to a .cdsenv File](#)

## viewsText

```
cdsLibManager.copy viewsText string "view_names"
```

### Description

Specifies a string of space-separated view names to copy or a valid filter string.

The default is " ".

### GUI Equivalent

None

### Examples

```
envGetVal("cdsLibManager.copy" "viewsText")  
envSetVal("cdsLibManager.copy" "viewsText" 'string " ")
```

### ***Related Topics***

[Using UNIX to Add Settings to a .cdsenv File](#)

## addPropFiles

```
cdsLibManager.copy addPropFiles boolean { t | nil }
```

### Description

Specifies whether to add dependent property files to a copy set.

The default is `t`.

### GUI Equivalent

Command	<i>Edit – Copy Preferences</i>
Form Field	<i>Library and Cell Property Files – Automatically add dependent property files to copy sets</i>

### Examples

```
envGetVal("cdsLibManager.copy" "addPropFiles")  
envSetVal("cdsLibManager.copy" "addPropFiles" 'boolean nil)
```

### ***Related Topics***

[Using UNIX to Add Settings to a .cdsenv File](#)



## **overwriteAll**

```
cdsLibManager.copyError overwriteAll boolean { t | nil }
```

### **Description**

Specifies whether overwrite is the selected action for all copy problems.

The default is `nil`.

### **GUI Equivalent**

None

### **Examples**

```
envGetVal("cdsLibManager.copyError" "overwriteAll")  
envSetVal("cdsLibManager.copyError" "overwriteAll" 'boolean t)
```

### ***Related Topics***

[Using UNIX to Add Settings to a .cdsenv File](#)

## addCellPropFiles

```
cdsLibManager.copyGlobals addCellPropFiles boolean { t | nil }
```

### Description

Specifies whether to add dependent cell property files to a copy set.

The default is `t`.

### GUI Equivalent

Command	<i>Edit – Copy Preferences</i>
Form Field	<i>Library and Cell Property Files – Include properties from – Cells</i>

### Examples

```
envGetVal("cdsLibManager.copyGlobals" "addCellPropFiles")  
envSetVal("cdsLibManager.copyGlobals" "addCellPropFiles" 'boolean t)
```

### Related Topics

[Using UNIX to Add Settings to a .cdsenv File](#)

## addLibPropFiles

```
cdsLibManager.copyGlobals addLibPropFiles boolean { t | nil }
```

### Description

Specifies whether to add dependent library property files to a copy set.

The default is `t`.

### GUI Equivalent

Command	<i>Edit – Copy Preferences</i>
Form Field	<i>Library and Cell Property Files – Libraries</i>

### Examples

```
envGetVal("cdsLibManager.copyGlobals" "addLibPropFiles")  
envSetVal("cdsLibManager.copyGlobals" "addLibPropFiles" 'boolean nil)
```

### ***Related Topics***

[Using UNIX to Add Settings to a .cdsenv File](#)

## expandRadio

```
cdsLibManager.copyGlobals expandRadio toggle (comanaged all)
```

### Description

Specifies what files are included of each cellview when performing copy operation. The choices are:

- `comanaged` means that include the comanaged files only of each cellview (the default)
- `all` means that include all the files of each cellview

### GUI Equivalent

Command	<i>Edit – Copy Preferences</i>
Form Field	<i>Cellview Contents</i>

### Examples

The following example returns the current value for the environment variable. The return value of `(t nil)` indicates that the `comanaged` option is enabled and `all` is disabled:

```
envGetVal("cdsLibManager.copyGlobals" "expandRadio")  
(t nil)
```

The example below enables `comanaged` and disables `all`:

```
envSetVal("cdsLibManager.copyGlobals" "expandRadio" 'toggle '(nil t))
```

### ***Related Topics***

[Copy Preferences Form](#)

[Using UNIX to Add Settings to a .cdsenv File](#)

## mpsRadio

```
cdsLibManager.copyGlobals mpsRadio toggle (remote local)
```

### Description

Specifies whether the copy operation uses the session's copy service (when available) or the Library Manager local copy engine only.

Specifies the copy service to be used for the copy operation.

- `remote` means that the copy service of the session is used for the copy operation (the default)
- `local` means that the copy service of the Library Manager local copy engine only is used for the copy operation

A special note about using the `ccpRegTrigger` SKILL function with reference to the `mpsRadio` environment variable setting:

- You can use the *Remote Copy Service* options on the Copy Preferences form to enable and disable user copy trigger execution.
- More advanced users can use the `ccpRegTrigger` SKILL function in the `.cdsinit` file to register a customized post-copy trigger function as follows:

```
procedure((copyTriggerPrint copyPhaseStr checkOffList supplementList
otherFromSpecs otherToSpecs updateList retHint ctxList reserved "stgggggggxx")
  let((retOK)
    retOK = t
    printf("Copy phase is \"%s\"\n" copyPhaseStr)
    printf("Calling options were %L\n", ctxList)
    printf("Pre-copy set is %L\n", checkOffList)
    printf("Post-copy is from %L\n", otherFromSpecs)
    printf("          to %L\n", otherToSpecs)
    retOK
  )
)
ccpRegTrigger("ccpPostCopyTrigger" 'copyTriggerPrint t)
```

You can remove this trigger using the `ccpRemoveTrigger` SKILL function as follows:

```
ccpRemoveTrigger("ccpPostCopyTrigger" 'copyTriggerPrint)
```

#### GUI Equivalent

Command	<i>Edit – Copy Preferences</i>
Form Field	<i>Remote Copy Service</i>

#### Examples

The following example returns the current value for the environment variable. The return value of `(t nil)` indicates that the `remote` option is enabled and `local` is disabled:

```
envGetVal("cdsLibManager.copyGlobals" "mpsRadio")  
(t nil)
```

The example below enables `local` and disables `remote`:

```
envSetVal("cdsLibManager.copyGlobals" "mpsRadio" 'toggle '(nil t))
```

#### ***Related Topics***

[Copy Preferences Form](#)

[Using UNIX to Add Settings to a .cdsenv File](#)

## useMonitor

```
cdsLibManager.copyGlobals useMonitor boolean { t | nil }
```

### Description

Specifies whether the progress monitor appears during a copy operation.

The default is `t`.

### GUI Equivalent

Command	<i>Edit – Copy Preferences</i>
Form Field	<i>Miscellaneous Settings – Enable file progress monitor</i>

### Examples

```
envGetVal("cdsLibManager.copyGlobals" "useMonitor")  
envSetVal("cdsLibManager.copyGlobals" "useMonitor" 'boolean nil)
```

### ***Related Topics***

[Using UNIX to Add Settings to a .cdsenv File](#)

[Copy Preferences Form](#)

## warnRenameDM

```
cdsLibManager.copyGlobals warnRenameDM boolean { t | nil }
```

### Description

Specifies whether a design management warning appears when matching items are renamed during a copy operation.

The default is `t`.

### GUI Equivalent

Command	<i>Edit – Copy Preferences</i>
Form Field	<i>Miscellaneous Settings – Warn about Rename of manage data (DM)</i>

### Examples

```
envGetVal("cdsLibManager.copyGlobals" "warnRenameDM")  
envSetVal("cdsLibManager.copyGlobals" "warnRenameDM" 'boolean nil)
```

### ***Related Topics***

[Using UNIX to Add Settings to a .cdsenv File](#)

[Copy Preferences Form](#)



## openView

```
cdsLibManager.copyVersion openView boolean { t | nil }
```

### Description

Specifies whether to open a cellview version after copying.

The default is `t`.

### GUI Equivalent

Command	<i>Design Manager – Version Info</i>
Form Field	<i>Copy Cellview Version – Open After Copy</i>

### Examples

```
envGetVal("cdsLibManager.copyVersion" "openView")  
envSetVal("cdsLibManager.copyVersion" "openView" 'boolean nil)
```

### ***Related Topics***

[Using UNIX to Add Settings to a .cdsenv File](#)

## toLibrary

```
cdsLibManager.copyVersion toLibrary string "destination_library_name"
```

### Description

Specifies the default destination library name. The default is " ".

### GUI Equivalent

Command	<i>Design Manager – Version Info</i>
Form Field	<i>Copy Cellview Version – To – Library</i>

### Examples

```
envGetVal("cdsLibManager.copyVersion" "toLibrary")  
envSetVal("cdsLibManager.copyVersion" "toLibrary" 'string "ahdLib")
```

### ***Related Topics***

[Using UNIX to Add Settings to a .cdsenv File](#)

## toView

```
cdsLibManager.copyVersion toView string "destination_view_name"
```

### Description

Specifies the default destination view name.

The default is " ".

### GUI Equivalent

Command	<i>Design Manager – Version Info</i>
Form Field	<i>Copy Cellview Version – To – View</i>

### Examples

```
envGetVal("cdsLibManager.copyVersion" "toLibrary")  
envSetVal("cdsLibManager.copyVersion" "toLibrary" 'string "adder")
```

### ***Related Topics***

[Using UNIX to Add Settings to a .cdsenv File](#)

## useOptionText

```
cdsLibManager.copyVersion useOptionText string "copy_options"
```

### Description

Specifies default copy options when copying a cellview version.

The default is " ".

### GUI Equivalent

Command	<i>Design Manager – Version Info</i>
Form Field	<i>Copy Cellview Version – Use Options</i>

### Examples

```
envGetVal("cdsLibManager.copyVersion" "useOptionText")  
envSetVal("cdsLibManager.copyVersion" "useOptionText" 'string  
"myOptionsFile.txt")
```

### ***Related Topics***

[Using UNIX to Add Settings to a .cdsenv File](#)

## useOptionsOn

```
cdsLibManager.copyVersion useOptionsOn boolean { t | nil }
```

### Description

Specifies whether copy options are enabled when copying a cellview version.

The default is `nil`.

### GUI Equivalent

Command	<i>Design Manager – Version Info</i>
Form Field	<i>Copy Cellview Version – Use Options</i>

### Examples

```
envGetVal("cdsLibManager.copyVersion" "useOptionsOn")  
envSetVal("cdsLibManager.copyVersion" "useOptionsOn" 'boolean t)
```

### ***Related Topics***

[Using UNIX to Add Settings to a .cdsenv File](#)

## addToCategoryName

```
cdsLibManager.copyWizard addToCategoryName string "category_name"
```

### Description

Specifies a default category name.

The default is " ".

### GUI Equivalent

Command	<i>Edit – Copy Wizard</i>
Form Field	<i>Add To Category</i>

### Examples

```
envGetVal("cdsLibManager.copyWizard" "addToCategoryName")  
envSetVal("cdsLibManager.copyWizard" "addToCategoryName" 'string "Transistors")
```

### ***Related Topics***

[Using UNIX to Add Settings to a .cdsenv File](#)

[Copy Wizard Form \(Simple Copy\)](#)

## addToCategoryOn

```
cdsLibManager.copyWizard addToCategoryOn boolean { t | nil }
```

### Description

Specifies whether the add-to-category option is enabled.

The default is `nil`.

### GUI Equivalent

Command	<i>Edit – Copy Wizard</i>
Form Field	<i>Add To Category</i>

### Examples

```
envGetVal("cdsLibManager.copyWizard" "addToCategoryOn")  
envSetVal("cdsLibManager.copyWizard" "addToCategoryOn" 'boolean t)
```

### ***Related Topics***

[Using UNIX to Add Settings to a .cdsenv File](#)

[Copy Wizard Form \(Simple Copy\)](#)

## addToCellsPattern

```
cdsLibManager.copyWizard addToCellsPattern string "t_filterString"
```

### Description

Specifies a filter string for matching a set of copied cells to add to a category.

The default is \*, indicates all copied cells.

### GUI Equivalent

Command	<i>Edit – Copy Wizard</i>
Form Field	<i>Cells</i>

### Examples

```
envGetVal("cdsLibManager.copyWizard" "addToCellsPattern")  
envSetVal("cdsLibManager.copyWizard" "addToCellsPattern" 'string "scurve")
```

### ***Related Topics***

[Using UNIX to Add Settings to a .cdsenv File](#)

[Copy Wizard Form \(Simple Copy\)](#)



## existenceCheck

```
cdsLibManager.copyWizard existenceCheck boolean { t | nil }
```

### Description

Specifies whether to select the Check existence in technology database check box in the Copy Wizard form by default.

The default is `nil`.

### GUI Equivalent

Command	<i>Edit – Copy Wizard</i>
Form Field	<i>Check existence in technology database</i>

### Examples

```
envGetVal ("cdsLibManager.copyWizard" "existenceCheck")  
envSetVal ("cdsLibManager.copyWizard" "existenceCheck" 'boolean t)
```

### ***Related Topics***

[Using UNIX to Add Settings to a .cdsenv File](#)

[Copy Wizard Form \(Simple Copy\)](#)

## extraViews

```
cdsLibManager.copyWizard extraViews string "extra_view_names"
```

### Description

Specifies extra view names to copy.

The default is " ".

### GUI Equivalent

Command	<i>Edit – Copy Wizard</i>
Form Field	<i>Exact Hierarchy – Extra Views</i>

### Examples

```
envGetVal("cdsLibManager.copyWizard" "extraViews")  
envSetVal("cdsLibManager.copyWizard" "extraViews" 'string "**")
```

### ***Related Topics***

[Using UNIX to Add Settings to a .cdsenv File](#)

[Copy Wizard Form \(Exact Hierarchy\)](#)

## **rerefCustomVias**

```
cdsLibManager.copyWizard rerefCustomVias boolean { t | nil }
```

### **Description**

Specifies whether to select the Re-reference Custom Via Defs check box in the Copy Wizard form by default.

The default is `nil`.

### **GUI Equivalent**

Command	<i>Edit – Copy Wizard</i>
Form Field	<i>Exact Hierarchy – Re-reference customViaDefs</i>

### **Examples**

```
envGetVal("cdsLibManager.copyWizard" "rerefCustomVias")  
envSetVal("cdsLibManager.copyWizard" "rerefCustomVias" 'boolean t)
```

### ***Related Topics***

[Using UNIX to Add Settings to a .cdsenv File](#)

[Copy Wizard Form \(Exact Hierarchy\)](#)

## updateChoice

```
cdsLibManager.copyWizard updateChoice toggle (entireLib copiesOnly)
```

### Description

Specifies what part of the library needs to be updated when updating instances.

- `entireLib` means to update the entire library when updating instances (the default)
- `copiesOnly` means to update the new copies only when updating instances

### GUI Equivalent

Command:     *Edit – Copy Wizard*

Field:        *Update Instances*

### Examples

The following example returns the current value for the environment variable. The return value of `(t nil)` indicates that the `entireLib` option is enabled and `copiesOnly` is disabled:

```
envGetVal("cdsLibManager.copyWizard" "updateChoice")  
(t nil)
```

The example below enables `copiesOnly` and disables `entireLib`:

```
envSetVal("cdsLibManager.copyWizard" "updateChoice" 'toggle '(nil t))
```

### ***Related Topics***

[Using UNIX to Add Settings to a .cdsenv File](#)

[Copy Wizard Form \(Exact Hierarchy\)](#)

## updateOn

```
cdsLibManager.copyWizard updateOn boolean { t | nil }
```

### Description

Specifies whether to update instances during the copy operation.

The default is `t`.

### GUI Equivalent

Command	<i>Edit – Copy Wizard</i>
Form Field	<i>Update Instances</i>

### Examples

```
envGetVal("cdsLibManager.copyWizard" "updateOn")  
envSetVal("cdsLibManager.copyWizard" "updateOn" 'boolean nil)
```

### ***Related Topics***

[Using UNIX to Add Settings to a .cdsenv File](#)

[Copy Wizard Form \(Exact Hierarchy\)](#)

## mapTimeout

```
cdsLibManager.customize mapTimeout float timeout_seconds
```

### Description

Specifies the timeout for mapCallbacks.

The default is 5 . 0.

### GUI Equivalent

None

### Examples

```
envGetVal("cdsLibManager.customize" "mapTimeout")  
envSetVal("cdsLibManager.customize" "mapTimeout" 'float 1.0)
```

### ***Related Topics***

[Using UNIX to Add Settings to a .cdsenv File](#)

## showDFIIWarning

```
cdsLibManager.customize showDFIIWarning boolean { t | nil }
```

### Description

Specifies whether a warning appears if a customization file is present but was not started from Virtuoso.

The default is `t`.

### GUI Equivalent

None

### Examples

```
envGetVal("cdsLibManager.customize" "showDFIIWarning")  
envSetVal("cdsLibManager.customize" "showDFIIWarning" 'boolean nil)
```

### *Related Topics*

[Using UNIX to Add Settings to a .cdsenv File](#)

## startupFile

```
cdsLibManager.customize startupFile string "startup_extension_file"
```

### Description

Specifies the name of the startup extension file.

The default is `cdsLibMgr.il`.

### GUI Equivalent

None

### Examples

```
envGetVal("cdsLibManager.customize" "startupFile")  
envSetVal("cdsLibManager.customize" "startupFile" 'string "myextensionfile.il")
```

### ***Related Topics***

[Using UNIX to Add Settings to a .cdsenv File](#)



## ddDb

```
cdsLibManager.database ddDb string "database"
```

### Description

Specifies the design database.

The default is `com.cadence.interfaces.libAccess.cddLib5xDatabase`.

### GUI Equivalent

None

### Examples

```
envGetVal("cdsLibManager.database" "ddDb")  
envSetVal("cdsLibManager.database" "ddDb" 'string "cddDatabase")
```

### ***Related Topics***

[Using UNIX to Add Settings to a .cdsenv File](#)

## **server**

```
cdsLibManager.database server string "server"
```

### **Description**

Specifies the database server.

The default is "com.cadence.interfaces.libAccess.ladLibraryServer".

### **GUI Equivalent**

None

### **Examples**

```
envGetVal("cdsLibManager.database" "server")  
envSetVal("cdsLibManager.database" "server" 'string "LibraryServer")
```

### ***Related Topics***

[Using UNIX to Add Settings to a .cdsenv File](#)

## fileRadio

```
cdsLibManager.defaults fileRadio toggle (Overwrite Merge values Retain values)
```

### Description

Specifies the default file save action of Library Manager. The choices are:

- `Overwrite` saves the values you type by overwriting your `.cdsenv` file
- `Merge values` saves the values you modify into your `.cdsenv` file (the default)
- `Retain values` saves the values you specify by creating another file

The default is `(nil t nil)`.

### GUI Equivalent

Command	<i>File – Save Defaults</i>
Form Field	<i>File Status</i>

### Examples

The following example returns the current value for the environment variable. The return value of `(nil t nil)` indicates that the `Merge values` option is enabled and the other two options are disabled:

```
envGetVal("cdsLibManager.defaults" "fileRadio")  
(nil t nil)
```

The example below enables `Overwrite` and `Retain values`, and disables the `Merge values` option:

```
envSetVal("cdsLibManager.defaults" "fileRadio" 'toggle '(t nil t))
```

### ***Related Topics***

[Using UNIX to Add Settings to a .cdsenv File](#)

[Save Library Manager Defaults Form](#)

## saveAllOn

```
cdsLibManager.defaults saveAllOn boolean { t | nil }
```

### Description

Specifies whether to save all possible values to the Library Manager defaults file.

The default is `nil`.

### GUI Equivalent

Command	<i>File – Save Defaults</i>
Form Field	<i>Options – All possible values</i>

### Examples

```
envGetVal("cdsLibManager.defaults" "saveAllOn")  
envSetVal("cdsLibManager.defaults" "saveAllOn" 'boolean t)
```

### ***Related Topics***

[Using UNIX to Add Settings to a .cdsenv File](#)

## saveAsText

```
cdsLibManager.defaults saveAsText string "save_as_name"
```

### Description

Specifies the default Save As name for the settings file. The default is `.cdsenv`.

### GUI Equivalent

Command	<i>File – Save Defaults</i>
Form Field	<i>Save Defaults File As</i>

### Examples

```
envGetVal("cdsLibManager.defaults" "saveAsText")  
envSetVal("cdsLibManager.defaults" "saveAsText" 'string "defaults")
```

### ***Related Topics***

[Using UNIX to Add Settings to a .cdsenv File](#)

## libCheckOn

```
cdsLibManager.delete libCheckOn boolean { t | nil }
```

### Description

Specifies whether to verify that a library is valid using its cdsinfo.tag file, prior to performing a delete operation to prevent deleting data or directories that are not in a valid library.

The default is `t`.

### GUI Equivalent

None

### Examples

```
envGetVal("cdsLibManager.delete" "libCheckOn")  
envSetVal("cdsLibManager.delete" "libCheckOn" 'boolean nil)
```

### *Related Topics*

[Using UNIX to Add Settings to a .cdsenv File](#)

## localRadio

```
cdsLibManager.delete localRadio toggle (both local)
```

### Description

Specifies what items are deleted.

- `both` means both local and inactive items are deleted from the Design Management system (the default)
- `local` means only the local items are deleted

### GUI Equivalent

Command	<i>Edit – Delete</i>
Form Field	<i>Options</i>

### Examples

The following example returns the current value for the environment variable. The return value of `(t nil)` indicates that `both` option is enabled and `local` is disabled:

```
envGetVal("cdsLibManager.delete" "localRadio")  
(t nil)
```

The example below enables `local` and disables `both`:

```
envSetVal("cdsLibManager.delete" "localRadio" 'toggle '(nil t))
```

### ***Related Topics***

[Delete Cells Form](#)

[Delete Libraries Form](#)

[Using UNIX to Add Settings to a .cdsenv File](#)

## regExpOn

```
cdsLibManager.delete regExpOn boolean { t | nil }
```

### Description

Specifies whether regular expressions are enabled for delete selection.

The default is `nil`.

### GUI Equivalent

None

### Examples

```
envGetVal("cdsLibManager.delete" "regExpOn")  
envSetVal("cdsLibManager.delete" "regExpOn" 'boolean t)
```

### ***Related Topics***

[Using UNIX to Add Settings to a .cdsenv File](#)



## overrideRadio

```
cdsLibManager.deleteTag overrideRadio toggle (no yes yesAll cancel)
```

### Description

Specifies the delete action when a library does not have the required `cdsinfo.tag` file and `cdsLibManager.delete libCheckOn` is set to `t`. The choices are:

- `no` means skip the specified library (the default)
- `yes` means delete the specified library
- `yesAll` means delete all the libraries
- `cancel` means cancel the delete operation

### GUI Equivalent

None

### Examples

The following example returns the current value for the environment variable. The return value of `(t nil nil nil)` indicates that the `no` option is enabled and the other three options are disabled:

```
envGetVal("cdsLibManager.deleteTag" "overrideRadio")  
(t nil nil nil)
```

The example below enables `yes` and `yesAll` and disables `no` and `cancel`:

```
envSetVal("cdsLibManager.deleteTag" "overrideRadio" 'toggle '(nil t t nil))
```

### *Related Topics*

[Using UNIX to Add Settings to a .cdsenv File](#)

## localRadio

```
cdsLibManager.deleteView localRadio toggle (both local)
```

### Description

Specifies what items are deleted.

- `both` means both local and inactive items are deleted from the Design Management system (the default)
- `local` means only the local items are deleted

### GUI Equivalent

Command	<i>Edit – Delete Cell Views</i>
Form Field	<i>Options</i>

### Examples

The following example returns the current value for the environment variable. The return value of `(t nil)` indicates that `both` option is enabled and `local` is disabled:

```
envGetVal("cdsLibManager.deleteView" "localRadio")  
(t nil)
```

The example below enables `local` and disables `both`:

```
envSetVal("cdsLibManager.deleteView" "localRadio" 'toggle '(nil t))
```

### Related Topics

[Delete Cell Views Form](#)

[Using UNIX to Add Settings to a .cdsenv File](#)

## viewFilterList

```
cdsLibManager.deleteView viewFilterList string "t_viewFilterList"
```

### Description

Specifies view names to delete.

The default is 'abstract', 'ahdl', 'autoLayout', 'behavior', 'cdsSpice', 'cmos\_sch', 'cmos.sch', 'functional', 'hpmns', 'hspiceS', 'layout', 'libra', 'mharm', 'schematic', 'spice', 'spectreS', 'symbol', 'system', 'verilog', 'verilogNetlist'.

You can specify any number of additional views using the viewFilterList variable with sequential numeric suffixes:

```
cdsLibManager.deleteView viewFilterList1 string ""  
cdsLibManager.deleteView viewFilterList2 string ""
```

### GUI Equivalent

None

### Examples

```
envGetVal("cdsLibManager.deleteView" "viewFilterList")  
envSetVal("cdsLibManager.deleteView" "viewFilterList" 'string "'symbol', 'system',  
'verilog', 'verilogNetlist'")
```

### *Related Topics*

[Using UNIX to Add Settings to a .cdsenv File](#)

## enableDmQuery

```
cdsLibManager.displayOptions enableDmQuery boolean { t | nil }
```

### Description

Enables the querying and retrieval of DM data so that state information can be shown for DM libraries. This information includes any data for the DM system if the extra columns are available.

The default is `t`.

### GUI Equivalent

Command	<i>View – Display Options</i>
Form Field	<i>State Analysis – Enable query of Design Management states</i>

### Examples

```
envGetVal("cdsLibManager.displayOptions" "enableDmQuery")  
envSetVal("cdsLibManager.displayOptions" "enableDmQuery" 'boolean nil)
```

### ***Related Topics***

[Using UNIX to Add Settings to a .cdsenv File](#)

## showExtendedStates

```
cdsLibManager.displayOptions showExtendedStates boolean { t | nil }
```

### Description

Allows DM tables to be shown in any viewing mode.

The default is `t`.

### GUI Equivalent

Command	<i>View – Display Options</i>
Form Field	<i>For Objects – View – Show extended states</i>

### Examples

```
envGetVal("cdsLibManager.displayOptions" "showExtendedStates")  
envSetVal("cdsLibManager.displayOptions" "showExtendedStates" 'boolean nil)
```

### ***Related Topics***

[Using UNIX to Add Settings to a .cdsenv File](#)

## autoModuleNameUpdate

```
ahdl autoModuleNameUpdate boolean { t | nil }
```

### Description

Updates the corresponding module name in the HDL file automatically while copying or renaming a text cellview.

The default is `t`. If this variable is set to `nil`, then after each copy or rename operation the application asks whether you want to update the module name automatically or not.

For example, if you rename functional text cellview `myVerilogCell`, with the module by the same name, to `myNewVerilogCell`, the module name updates in the Verilog file automatically.

### GUI Equivalent

None

### Examples

```
envGetVal("ahdl" "autoModuleNameUpdate")  
envSetVal("ahdl" "autoModuleNameUpdate" 'boolean nil)
```

### ***Related Topics***

[Text Cellviews](#)

## **matchModuleNameCellName**

```
ahdl autoModuleNameUpdate string "status"
```

### **Description**

Specifies if there is a mismatch between the cell name and module name.

The default value is `ignore`.

### **GUI Equivalent**

None

### **Example**

```
envGetVal("ahdl" "matchModuleNameCellName")  
envSetVal("ahdl" "matchModuleNameCellName" 'string "error")
```

### ***Related Topics***

[Text Cellviews](#)

## sortCellNameCompareFn

```
designEditor.fileSpec sortCellNameCompareFn string "alphanumeric_order"
```

### Description

Displays the cells in the *Cell* list box of the Open File form using the same sort of alphanumeric order as used in Library Manager.

The default value is `alphalessp`.

### GUI Equivalent

None

### Example

```
envGetVal("designEditor.fileSpec" "sortCellNameCompareFn")
envSetVal("designEditor.fileSpec" "sortCellNameCompareFn" 'string
"naturalStrLessp")
```

### ***Related Topics***

[Opening the Library Browser Form](#)



## libSelectCellViewCombineMode

```
cdsLibManager.filter libSelectCellViewCombineMode cyclic { "never" | "always" }
```

### Description

Controls the cell filter mode of Library Browser form.

The default is `never`.

You can choose to make the cell filtering faster by preventing the view filter from being applied to it or you can choose to make it more accurate but slightly slower by letting the view filter affect the list of cells displayed.

**Note:** If you access the Library Browser form from Virtuoso, place your `.cdsenv` in your `$HOME` directory; unlike the Library Manager, Virtuoso does not look for the `.cdsenv` file in your current working directory.

### Arguments

<code>never</code>	Applies cell filter before the list of cells is displayed when a library in the Library Browser form is selected, and the view filter gets ignored, which means that even cells that do not contain any views matching the view filter are displayed. As a result, the list of cells is displayed much faster. <code>"never"</code> is the default setting.
<code>always</code>	Applies both the cell filter and view filter before the list of cells is displayed when a library in the Library Browser form is selected, only cells containing views that match the view filter are displayed. With this setting, the result takes longer to display the list of cells in the library.

### GUI Equivalent

None

### Example

```
envGetVal("cdsLibManager.filter" "libSelectCellViewCombineMode")  
envSetVal("cdsLibManager.filter" "libSelectCellViewCombineMode" 'cyclic "always")
```

## **Cadence Library Manager User Guide**

### Library Manager Environment Variables

---

#### ***Related Topics***

[Library Browser Form](#)

## skipLibsText

```
cdsLibManager.copyWizard skipLibsText string "lib1 lib2"
```

### Description

Specifies names of libraries to skip during the copy operation, when skipLibsOn is `t`. The default is to use the generated skip list, `"*"`. The generated skip list consists of all defined libraries in `cds.lib` except the one containing the source item.

If exceeding the `.cdsenv` file line length limit becomes a problem, you can specify additional skip libraries using additional `skipLibsText` variables with sequential numeric suffixes as follows:

```
cdsLibManager.copy skipLibsText1 string ""  
cdsLibManager.copy skipLibsText2 string ""
```

### GUI Equivalent

None

### Example

```
envGetVal("cdsLibManager.copyWizard" "skipLibsText")  
envSetVal("cdsLibManager.copyWizard" "skipLibsText" 'string "cdsDefTechLib")
```

### *Related Topics*

[Using UNIX to Add Settings to a .cdsenv File](#)

## dmSyncDelay

`cdsLibManager.displayOptions dmSyncDelay float delay_period_seconds`

### Description

Specifies the number of seconds to set for the delay period, when the feature is enabled.

The default is 2 . 5.

### GUI Equivalent

None

### Example

```
envGetVal("cdsLibManager.displayOptions" "dmSyncDelay")
envSetVal("cdsLibManager.displayOptions" "dmSyncDelay" 'float 1.0)
```

### ***Related Topics***

[Display Options Form](#)

## enableDmSyncDelay

```
cdsLibManager.displayOptions enableDmSyncDelay boolean { t | nil }
```

### Description

Allows you to enable or disable the DM Sync feature.

The default is `nil`.

### GUI Equivalent

None

### Examples

```
envGetVal("cdsLibManager.displayOptions" "enableDmSyncDelay")  
envSetVal("cdsLibManager.displayOptions" "enableDmSyncDelay" 'boolean t)
```

### ***Related Topics***

[Display Options Form](#)

## showHiddenLibraries

```
cdsLibManager.displayOptions showHiddenLibraries boolean { t | nil }
```

### Description

Specifies whether libraries that are hidden because of a display attribute set on them should be hidden or displayed.

The default is `nil`.

### GUI Equivalent

Command	<i>View – Display Options</i>
Form Field	<i>Show hidden libraries</i>

### Examples

```
envGetVal("cdsLibManager.displayOptions" "showHiddenLibraries")  
envSetVal("cdsLibManager.displayOptions" "showHiddenLibraries" 'boolean t)
```

### ***Related Topics***

[Display Options Form](#)

## showLibraryColors

```
cdsLibManager.displayOptions showLibraryColors boolean { t | nil }
```

### Description

Specifies whether libraries should be displayed in custom colors.

The default is `t`.

### GUI Equivalent

Command	<i>View – Display Options</i>
Form Field	<i>Show library colors</i>

### Examples

```
envGetVal("cdsLibManager.displayOptions" "showLibraryColors")  
envSetVal("cdsLibManager.displayOptions" "showLibraryColors" 'boolean nil)
```

### ***Related Topics***

[Display Options Form](#)

## showLibraryCustomIcons

```
cdsLibManager.displayOptions showLibraryCustomIcons boolean { t | nil }
```

### Description

Specifies whether custom library icons should be displayed next to libraries in both the Tree view and Lists view.

The default is `t`.

### GUI Equivalent

Command	<i>View – Display Options</i>
Form Field	<i>Show custom library</i>

### Examples

```
envGetVal("cdsLibManager.displayOptions" "showLibraryCustomIcons")  
envSetVal("cdsLibManager.displayOptions" "showLibraryCustomIcons" 'boolean nil)
```

### ***Related Topics***

[Display Options Form](#)



## showListViewIcons

```
cdsLibManager.displayOptions showListViewIcons boolean { t | nil }
```

### Description

Specifies whether custom library icons should be displayed next to libraries in the Lists view.

The default is `t`.

### GUI Equivalent

Command	<i>View – Display Options</i>
Form Field	<i>Show Lists view library icons</i>

### Examples

```
envGetVal("cdsLibManager.displayOptions" "showListViewIcons")  
envSetVal("cdsLibManager.displayOptions" "showListViewIcons" 'boolean nil)
```

### ***Related Topics***

[Display Options Form](#)

## cellFilter

```
cdsLibManager.filter cellFilter string "default_string"
```

### Description

Specifies the default cell filter string. The default is " ".

### GUI Equivalent

None

### Examples

```
envGetVal("cdsLibManager.filter" "cellFilter")  
envSetVal("cdsLibManager.filter" "cellFilter" 'string "bsource")
```

### ***Related Topics***

[Using UNIX to Add Settings to a .cdsenv File](#)

## viewFilter

```
cdsLibManager.filter viewFilter string "default_string"
```

### Description

Specifies the default view filter string. The default is " ".

### GUI Equivalent

None

### Examples

```
envGetVal("cdsLibManager.filter" "viewFilter")  
envSetVal("cdsLibManager.filter" "viewFilter" 'string "symbol")
```

### ***Related Topics***

[Using UNIX to Add Settings to a .cdsenv File](#)

## addHostID

```
cdsLibManager.log addHostID boolean { t | nil }
```

### Description

Specifies whether the host name is part of the log file name. The default is `nil`.

The log file name format is

```
baseName[.userID][.hostName][.PID][.sequenceNumber].log.
```

### GUI Equivalent

None

### Examples

```
envGetVal("cdsLibManager.log" "addHostID")  
envSetVal("cdsLibManager.log" "addHostID" 'boolean t)
```

### *Related Topics*

[Using UNIX to Add Settings to a .cdsenv File](#)

## addProcID

```
cdsLibManager.log addProcID boolean { t | nil }
```

### Description

Specifies whether the process ID is part of the log file name. The default is `nil`.

The log file name format is

```
baseName [.userID] [.hostName] [.PID] [.sequenceNumber] .log.
```

### GUI Equivalent

None

### Examples

```
envGetVal("cdsLibManager.log" "addProcID")  
envSetVal("cdsLibManager.log" "addProcID" 'boolean t)
```

### *Related Topics*

[Using UNIX to Add Settings to a .cdsenv File](#)

## addSequence

```
cdsLibManager.log addSequence boolean { t | nil }
```

### Description

Specifies whether a sequence number is part of the log file name.

The default is `nil`.

The log file name format is

```
baseName[.userID][.hostName][.PID][.sequenceNumber].log.
```

### GUI Equivalent

None

### Examples

```
envGetVal("cdsLibManager.log" "addSequence")  
envSetVal("cdsLibManager.log" "addSequence" 'boolean t)
```

### ***Related Topics***

[Using UNIX to Add Settings to a .cdsenv File](#)

## addUserID

```
cdsLibManager.log addUserID boolean { t | nil }
```

### Description

Specifies whether the user ID is part of the log file name. The default is `nil`.

The log file name format is

```
baseName[.userID][.hostName][.PID][.sequenceNumber].log .
```

### GUI Equivalent

None

### Examples

```
envGetVal("cdsLibManager.log" "addUserID")  
envSetVal("cdsLibManager.log" "addUserID" 'boolean t)
```

### *Related Topics*

[Using UNIX to Add Settings to a .cdsenv File](#)

## baseName

```
cdsLibManager.log baseName string "baseName"
```

### Description

Specifies the base name of the Library Manager log file. The program appends the location you specify to the CDS\_LOG\_PATH location.

The default is libManager.

### GUI Equivalent

None

### Examples

```
envGetVal("cdsLibManager.log" "baseName")  
envSetVal("cdsLibManager.log" "baseName" 'string "/mylibs/libManager")
```

### *Related Topics*

[Using UNIX to Add Settings to a .cdsenv File](#)



## categoryText

```
cdsLibManager.main categoryText string "default_category_name"
```

### Description

Specifies a default category name for the main Library Manager window.

The default is " ".

### GUI Equivalent

None

### Examples

```
envGetVal("cdsLibManager.main" "categoryText")  
envSetVal("cdsLibManager.main" "categoryText" 'string "Pins")
```

### ***Related Topics***

[Using UNIX to Add Settings to a .cdsenv File](#)

## cellLevelText

```
cdsLibManager.main cellLevelText string "default_cell_name"
```

### Description

Specifies a default cell name for the main Library Manager window.

The default is " ".

### GUI Equivalent

None

### Examples

```
envGetVal("cdsLibManager.main" "cellLevelText")  
envSetVal("cdsLibManager.main" "cellLevelText" 'string "ccvs")
```

### ***Related Topics***

[Using UNIX to Add Settings to a .cdsenv File](#)

## **dblClickEditCellView**

```
cdsLibManager.main dblClickEditCellView boolean { t | nil }
```

### **Description**

Specifies whether the double-click action on a view name in the main Library Manager window opens a cellview for editing. If set to `nil`, the double-click action on a view name in the main Library Manager window opens a cellview in the Read-only mode.

The default is `t`.

### **GUI Equivalent**

None

### **Examples**

```
envGetVal("cdsLibManager.main" "dblClickEditCellView")  
envSetVal("cdsLibManager.main" "dblClickEditCellView" 'boolean nil)
```

### ***Related Topics***

[Using UNIX to Add Settings to a .cdsenv File](#)

## libraryText

```
cdsLibManager.main libraryText string "default_library_name"
```

### Description

Specifies a default library name for the main Library Manager window.

The default is " ".

### GUI Equivalent

None

### Examples

```
envGetVal("cdsLibManager.main" "libraryText")  
envSetVal("cdsLibManager.main" "libraryText" 'string "tutorial")
```

### ***Related Topics***

[Using UNIX to Add Settings to a .cdsenv File](#)

## showCategoriesOn

```
cdsLibManager.main showCategoriesOn boolean { t | nil }
```

### Description

Specifies whether categories appear in Lists mode.

The default is `nil`.

### GUI Equivalent

Command	<i>Library Manager</i>
Form Field	<i>Show Categories</i>

### Examples

```
envGetVal("cdsLibManager.main" "showCategoriesOn")  
envSetVal("cdsLibManager.main" "showCategoriesOn" 'boolean t)
```

### ***Related Topics***

[Library Manager Form](#)

## showFilesOn

```
cdsLibManager.main showFilesOn boolean { t | nil }
```

### Description

Specifies whether files appear in Lists mode.

The default is `nil`.

### GUI Equivalent

Command	<i>Library Manager</i>
Form Field	<i>Show Files</i>

### Examples

```
envGetVal("cdsLibManager.main" "showFilesOn")  
envSetVal("cdsLibManager.main" "showFilesOn" 'boolean t)
```

### ***Related Topics***

[Library Manager Form](#)

## showNonVirtuosoViewtypes

```
cdsLibManager.main showNonVirtuosoViewtypes boolean { t | nil }
```

### Description

Displays non-Virtuoso view types for the selected cell.

The default is `t`.

### GUI Equivalent

Command	<i>File – Open – Browse – Library Browser</i>
Form Field	<i>Show Non-Virtuoso View Types</i>

### Examples

```
envGetVal("cdsLibManager.main" "showNonVirtuosoViewtypes")  
envSetVal("cdsLibManager.main" "showNonVirtuosoViewtypes" 'boolean nil)
```

### ***Related Topics***

[Library Browser Form](#)

[Opening the Library Browser Form](#)

## viewLevelText

```
cdsLibManager.main viewLevelText string "view_name"
```

### Description

Specifies a default view name for the main Library Manager windows.

The default is " ".

### GUI Equivalent

None

### Examples

```
envGetVal("cdsLibManager.main" "viewLevelText")  
envSetVal("cdsLibManager.main" "viewLevelText" 'string "tutorial")
```

### ***Related Topics***

[Library Manager Form](#)



## dmRadio

```
cdsLibManager.newLib dmRadio toggle (default noDM)
```

### Description

Specifies whether to use design management for a newly created library.

- `default` means use design management for the new library (the default)
- `noDM` means do not use design management for the new library

### GUI Equivalent

Command	<i>File – New Library</i>
Form Field	<i>Design Manager</i>

### Examples

The following example returns the current value for the environment variable. The return value of `(t nil)` indicates that the `default` option is enabled and `noDM` is disabled:

```
envGetVal("cdsLibManager.newLib" "dmRadio")  
(t nil)
```

The example below enables `noDM` and disables `default`:

```
envSetVal("cdsLibManager.newLib" "dmRadio" 'toggle '(nil t))
```

### ***Related Topics***

[New Library Form](#)

## pathText

```
cdsLibManager.newLib pathText string "path_string"
```

### Description

Specifies a default path string for creating a new library.

The default is " ".

### GUI Equivalent

None

### Examples

```
envGetVal("cdsLibManager.newLib" "pathText")  
envSetVal("cdsLibManager.newLib" "pathText" 'string "mypath.txt")
```

### ***Related Topics***

[New Library Form](#)

## windowBehavior

```
cdsLibManager.open windowBehavior cyclic { "newWindow" | "raiseExisting" }
```

### Description

Specifies whether an application always opens a cellview in a new window or brings up a window that already has the cellview open.

The default value is `newWindow`. In this case, the application opens the cellview in a new window.

When the value is `raiseExisting`, the application brings up the window that already has the cellview open, with the cellview tab as the active one.

### GUI Equivalent

None

### Examples

```
envGetVal("cdsLibManager.open" "windowBehavior")  
envSetVal("cdsLibManager.open" "windowBehavior" 'cyclic "raiseExisting")
```

### *Related Topics*

[Opening a Cellview](#)

## **forceEnv**

```
cdsLibManager.option forceEnv boolean { t | nil }
```

### **Description**

Specifies whether dialogs use `.cdsenv` settings instead of programmed default values.

The default is `nil`.

### **GUI Equivalent**

None

### **Examples**

```
envGetVal("cdsLibManager.option" "forceEnv")  
envSetVal("cdsLibManager.option" "forceEnv" 'boolean t)
```

### ***Related Topics***

[Using UNIX to Add Settings to a .cdsenv File](#)

## useDMfilter

```
cdsLibManager.option useDMfilter boolean { t | nil }
```

### Description

Specifies whether the Design Manager menu commands are sensitive to the current state of the selected items in the design management system.

The default is `t`.

If you set the design management state check to `nil`, the Design Manager menu commands are always active as long as the library is managed by a working design management system). Turning off the state check might help if your Design Manager menus and commands are slow to respond because your design management files are located across a slow network, or state queries are made using a slow network transport.

### GUI Equivalent

None

### Examples

```
envGetVal("cdsLibManager.option" "useDMfilter")  
envSetVal("cdsLibManager.option" "useDMfilter" 'boolean nil)
```

### ***Related Topics***

[Using UNIX to Add Settings to a .cdsenv File](#)

## useFastDM

```
cdsLibManager.option useFastDM boolean { t | nil }
```

### Description

Specifies whether to use fast design management querying. The default is `t`.

Fast design management (DM) querying involves checking a `master.tag` file instead of the master file. It is typically much faster to query the much smaller, `master.tag` file of a cellview.

### GUI Equivalent

None

### Examples

```
envGetVal("cdsLibManager.option" "useFastDM")  
envSetVal("cdsLibManager.option" "useFastDM" 'boolean nil)
```

### *Related Topics*

[Using UNIX to Add Settings to a .cdsenv File](#)

## updateOn

```
cdsLibManager.rename updateOn boolean { t | nil }
```

### Description

Specifies whether to update instances when renaming an item.

The default is `t`.

### GUI Equivalent

None

### Examples

```
envGetVal("cdsLibManager.rename" "updateOn")  
envSetVal("cdsLibManager.rename" "updateOn" 'boolean nil)
```

### ***Related Topics***

[Using UNIX to Add Settings to a .cdsenv File](#)

## refreshSessionOn

```
cdsLibManager.renameRefLib refreshSessionOn boolean { t | nil }
```

### Description

Specifies whether to refresh the session after renaming a reference library.

The default is `t`.

### GUI Equivalent

Command	<i>Edit – Rename Reference Library</i>
Form Field	<i>Refresh Session</i>

### Examples

```
envGetVal("cdsLibManager.renameRefLib" "refreshSessionOn")  
envSetVal("cdsLibManager.renameRefLib" "refreshSessionOn" 'boolean nil)
```

### ***Related Topics***

[Rename Reference Library Form](#)



## useNameOn

```
cdsLibManager.submit useNameOn boolean { t | nil }
```

### Description

Specifies whether the submit request name is enabled.

The default is `nil`.

### GUI Equivalent

Command	<i>Design Manager – Submit</i>
Form Field	<i>Submit Options – Request Name</i>

### Examples

```
envGetVal("cdsLibManager.submit" "useNameOn")  
envSetVal("cdsLibManager.submit" "useNameOn" 'boolean t)
```

### ***Related Topics***

[Submit Form](#)

## useNameText

```
cdsLibManager.submit useNameText string "default_request_name"
```

### Description

Specifies a default submit request name. The default is " ".

### GUI Equivalent

Command	<i>Design Manager – Submit</i>
Form Field	<i>Submit Options – Request Name</i>

### Examples

```
envGetVal("cdsLibManager.submit" "useNameText")  
envSetVal("cdsLibManager.submit" "useNameText" 'string "myfirstrequest")
```

### ***Related Topics***

[Submit Form](#)

## useOptionText

```
cdsLibManager.submit useOptionText string "submit_options"
```

### Description

Specifies submit options. The default is " ".

### GUI Equivalent

Command	<i>Design Manager – Submit</i>
Form Field	<i>Submit Options – Use Options</i>

### Examples

```
envGetVal("cdsLibManager.submit" "useOptionText")  
envSetVal("cdsLibManager.submit" "useOptionText" 'string "myOptionsFile.txt")
```

### ***Related Topics***

[Submit Form](#)

## useOptionsOn

```
cdsLibManager.submit useOptionsOn boolean { t | nil }
```

### Description

Specifies whether submit options are enabled. The default is `nil`.

### GUI Equivalent

Command	<i>Design Manager – Submit</i>
Form Field	<i>Submit Options – Use Options</i>

### Examples

```
envGetVal("cdsLibManager.submit" "useOptionsOn")  
envSetVal("cdsLibManager.submit" "useOptionsOn" 'boolean t)
```

### ***Related Topics***

[Submit Form](#)

## useNameOn

```
cdsLibManager.update useNameOn boolean { t | nil }
```

### Description

Specifies whether the update from name is enabled.

The default is `nil`.

### GUI Equivalent

None

### Examples

```
envGetVal("cdsLibManager.update" "useNameOn")  
envSetVal("cdsLibManager.update" "useNameOn" 'boolean t)
```

### ***Related Topics***

[Using UNIX to Add Settings to a .cdsenv File](#)

## useOptionsOn

```
cdsLibManager.update useOptionsOn boolean { t | nil }
```

### Description

Specifies whether update options are enabled.

The default is `nil`.

### GUI Equivalent

None

### Examples

```
envGetVal("cdsLibManager.update" "useOptionsOn")  
envSetVal("cdsLibManager.update" "useOptionsOn" 'boolean t)
```

### ***Related Topics***

[Using UNIX to Add Settings to a .cdsenv File](#)

## compressionOn

```
cdsLibManager.newLib compressionOn boolean { t | nil }
```

### Description

Specifies whether the Compression enabled check box in the New Library form should remain selected or deselected by default.

The default is `nil`, which means that the check box is deselected by default.

### GUI Equivalent

Command	<i>Library Manager – File</i>
Form Field	<i>New Library – Compression enabled</i>

### Examples

```
envGetVal("cdsLibManager.newLib" "compressionOn")  
envSetVal("cdsLibManager.newLib" "compressionOn" 'boolean t)
```

### ***Related Topics***

[New Library Form](#)

## enable

```
ui.thumbnails enable boolean { t | nil }
```

### Description

Generates thumbnails. The default is `t`.

If `enable` is set to `nil`, thumbnails does not get displayed or generated. This variable also overrides the `display` and `generate` variable settings

### GUI Equivalent

None

### Example

```
envGetVal("ui.thumbnails" "enable")  
envSetVal("ui.thumbnails" "enable" 'boolean nil)
```

### ***Related Topics***

[Library Manager Toolbar](#)



## display

```
ui.thumbnails display boolean { t | nil }
```

### Description

Disables thumbnail display in both the Library Manager and *Open* forms.

The default is `t`.

### GUI Equivalent

None

### Examples

```
envGetVal("ui.thumbnails" "display")  
envSetVal("ui.thumbnails" "display" 'boolean nil)
```

### ***Related Topics***

[Library Manager Toolbar](#)

## **generate**

```
ui.thumbnails generate boolean { t | nil }
```

### **Description**

Disables auto-generation of thumbnails. The default is `t`.

### **GUI Equivalent**

None

### **Examples**

```
envGetVal("ui.thumbnails" "generate")  
envSetVal("ui.thumbnails" "generate" 'boolean nil)
```

### ***Related Topics***

[Library Manager Toolbar](#)

## **verbose**

```
ui.thumbnails verbose boolean { t | nil }
```

### **Description**

Turns on verbose mode which prints the location of a saved file when a thumbnail is saved. The default is `t`.

### **GUI Equivalent**

None

### **Examples**

```
envGetVal("ui.thumbnails" "verbose")  
envSetVal("ui.thumbnails" "verbose" 'boolean nil)
```

### ***Related Topics***

[Library Manager Toolbar](#)

## **Cadence Library Manager User Guide**

### **Library Manager Environment Variables**

---

---

## Library Manager Forms

---

- [Access Permission Form](#)
- [Add Property Form](#)
- [Attach Library to Technology Library Form](#)
- [Cancel Check Out Form](#)
- [Cell Property Editor Form](#)
- [Change Library References Form](#)
- [Check In Form](#)
- [Check Out Form](#)
- [Choose Environment File To Load Form](#)
- [Copy Cell File Form](#)
- [Copy Cell Form](#)
- [Copy Cellview Version Form](#)
- [Copy Library File Form](#)
- [Copy Library Form](#)
- [Copy Preferences Form](#)
- [Copy View Form](#)
- [Copy Wizard Form \(Simple Copy\)](#)
- [Copy Wizard Form \(Hierarchical\)](#)
- [Copy Wizard Form \(Exact Hierarchy\)](#)
- [Copy Wizard Form \(By View\)](#)
- [Copy Wizard Form \(By Configuration\)](#)

## Cadence Library Manager User Guide

### Library Manager Forms

---

- [Delete By View Form](#)
- [Delete Cells Form](#)
- [Delete Cell Views Form](#)
- [Delete Libraries Form](#)
- [Delete Library Views Form](#)
- [Display Options Form](#)
- [Display Settings Form](#)
- [DM File Status Form](#)
- [New Category Form](#)
- [Library Browser Form](#)
- [Library Manager Form](#)
- [Library Property Editor Form](#)
- [Load Technology File Form](#)
- [Modify 'propertyName' Form](#)
- [New Category Form](#)
- [New File Form](#)
- [New Library Form](#)
- [Reference Existing Technology Libraries Form](#)
- [Rename Cell Form](#)
- [Rename File Form](#)
- [Rename Library Form](#)
- [Rename Reference Library Form](#)
- [Rename View Form](#)
- [Save Library Manager Defaults Form](#)
- [Select an icon Form](#)
- [Submit Form](#)
- [Technology File for New Library Form](#)

## Cadence Library Manager User Guide

### Library Manager Forms

---

- [Version Information Form](#)
- [View Property Editor Form](#)
- [View Property Editor Form](#)

## Access Permission Form

Use the Access Permission Form to check the permissions given to the owner, group, and others.

Field	Description
<b><i>Library functional</i></b>	This section displays the information of the file owner and the group.
<i>Owner</i>	Displays the user ID of the file owner.
<i>Group</i>	Displays the name of the working group to which the owner is assigned by the project leader or system administrator.
	This section displays the read, write, and execute permission given to the following categories:
<i>Owner</i>	Displays the access permissions given to the file owner.  By default, the owner has all the access permissions, and the owner must have write permission before the other two categories can have write permission.
<i>Group</i>	Displays the access permissions given to the working group of the owner.
<i>Others</i>	Displays the access permissions given to anyone who has access to the files.
<i>Apply</i>	Sets the values you selected in the <i>Access Permission</i> group box.

**Note:** If the Group or Others category have the write or execute permission then by default they must also have read permission.

### ***Related Topics***

[Library Display Settings](#)

[Viewing and Changing File Permissions](#)



## Add Property Form

Use the Add Property Form to modify the properties of the cellview.

Field	Description
<i>Name</i>	Lets you assign a name to the property.
<i>Type</i>	Lets you choose a property type. Depending on the type you choose, the remaining fields change to prompt you for values for the property.
<i>Value</i>	Displays the default value for the property.
<i>Minimum Value</i>	Displays the minimum value for the property.
<i>Maximum Value</i>	Displays the maximum value for the property.  Used for <i>int</i> , <i>float</i> , and time property types. The View Property Editor form or the Cell Property Editor form displays these minimum and maximum values next to the property name.
<i>Defaults</i>	Clears all values in the fields and sets <i>Type</i> to <i>int</i> .

### ***Related Topics***

[Adding Properties to a Library, Cell, or View](#)

## **Attach Library to Technology Library Form**

Use this form to attach new library to the available technology libraries

---

<b>Field</b>	<b>Description</b>
<i>New Library</i>	Displays the name of the new library to be created
<i>Technology Library</i>	Lists the available technology libraries that can be chosen to be attached to the new library.

---

### ***Related Topics***

[Attaching a New Library to an Existing Technology Library](#)

## Cancel Check Out Form

Use this to cancel a check out operation for a library, cell, view, or file.

**Note:** If another user in your workarea has locked a file, you cannot cancel the check-out. As a result, the Cancel Checkout form does not get displayed.

If you locked a file, the software prompts you to confirm that you do not want to save any changes.

When you cancel a check-out operation, the software restores your workarea and the project design management repository to the states they were in prior to the check-out. You can cancel a check-out operation if you have not made any changes to the checked-out files or do not want to save any changes you made to checked-out files.

Cadence recommends that you do not cancel check-out for design data files that are locked. A locked design data file is one that someone else is currently editing. If you try to cancel check-out for a locked file, an error message gets displayed. To resolve the error, close design windows or change the files to read-only mode before you cancel check-out.

The software cancels the check-out operation for each selected file. Files that are not selected remain checked out.

Field	Description
<b><i>Files to Cancel Check Out</i></b>	This section selects the files in which you want to cancel the check out operation.
<i>Refresh Status</i>	Refreshes the current status of the items available in the form.
<i>Show</i>	Selects the library files need to be displayed.
<i>Select All</i>	Adds all files to the selection set.
<i>Deselect All</i>	Remove all files from the selection set.  For individual files, deselect the check box to the left of the file name for each file you want to remove from the selection set for canceling check-out.
<i>Invert All</i>	Toggles each checked box to the opposite checked setting.
<b><i>Cancel Check Out Options</i></b>	This section displays the Cancel Checkout form format same as the Check In form and the Cancel Out form.

## Cadence Library Manager User Guide

### Library Manager Forms

---

Field	Description
	<p>However, the ability to switch command tabs, in the respective forms, is restricted to the command action that is currently applicable, as the legal file sets are mutually exclusive.</p> <p>The only exception is in relation to the <i>Cancel Checkout</i> and <i>Check In</i> tabs, when using the <i>Check Out</i> command, as either of these actions could be applied to checked out files.</p>
<i>Use Options</i>	Enters any check-out cancellation options specific to the particular design management system you want to use.

#### ***Related Topics***

[Selecting Items for Copying in the Copy Wizard](#)

[Files in Read-Only Mode](#)

## Cell Property Editor Form

Use this form to edit the properties of the selected cellview.

Fields	Description
<b>Cell</b>	This section displays the information of the selected cellview.
<i>Name</i>	Displays the cell name of the selected cellview.
<i>Owner</i>	Displays the user ID of the owner of the selected cellview.
<i>Group</i>	Displays the group of the owner of the selected cellview.
<i>Last Modify</i>	Displays the date and time of the last modification to the cellview.
<i>Read Path</i>	Displays the path to a read-only version of the cellview.
<i>Write Path</i>	Displays the path to a writable version of the cellview.
<b>UNIX Permissions Mode</b>	This section displays the read/write/execute permissions for various authorities. It is not allowed to change any of the values in the UNIX Permissions Mode group box.
<i>Owner</i>	Displays the read/write/execute permissions for the owner of the cellview.
<i>Group</i>	Displays the read/write/execute permissions for the group of the owner of the cellview.
<i>Other</i>	Displays the read/write/execute permissions for anyone who has access to the cellview.
<b>Properties</b>	Lists all the properties available in the specified cell at the bottom of the Cell Property Editor form and you can edit them on this form.  If the specified cell has no properties then <i>No Property Attached</i> text is displayed.
<i>Add</i>	Opens the Add Property form.
<i>Delete</i>	Removes the selected property.
<i>Modify</i>	Opens the Modify <i>propertyName</i> form.

### ***Related Topics***

[Modify 'propertyName' Form](#)

[Library Property Editor Form](#)

## **Cadence Library Manager User Guide**

### **Library Manager Forms**

---

[View Property Editor Form](#)

## Change Library References Form

Use this form to change the name of the reference library.

Fields	Description
<i>In Library</i>	Displays the name of the design library that uses a reference library whose name you want to change.
<i>From Library</i>	Displays the name of the current reference library/libraries.
<i>To Library</i>	Displays the name of the new reference library.
<i>Show Selected Libraries Only</i>	Shows only the selected libraries in <i>In Library</i> , <i>From Library</i> , and <i>To Library</i> list box.
<i>Add</i>	Displays all the selected libraries in the <i>In Library</i> , <i>From Library</i> , and <i>To Library</i> list box in the <i>Change List</i> section.
<i>Move Up</i>	Moves the entry up the order in the Change List section.
<i>Move Down</i>	Moves the entry down the order in the Change List section.
<i>Delete</i>	Deletes the entry from the Change List section.

### ***Related Topics***

[Changing Library Reference](#)

## Check In Form

Use this form to check in a design.

Both the check-in and check-out processes control access to the design data files that design team members store in a project design management repository, so that:

- When you check out a file, the software copies the file from the project design management repository to your current workarea.
- When you check in a file, the software copies the file, as a completed version, from your workarea to the project design management repository and assigns the next version number.

**Note:** You can optimize the performance of check-in operations by setting the `DD_GDM_OPTIMIZE` environment variable to `yes` before you start Virtuoso.

Cadence recommends that you do not check in design data files that are locked. A locked design data file is one that someone else is currently editing. If you try to check in a locked file, an error message gets displayed. To resolve the error, close design windows or change the files to read-only mode before you check them in.

---

Field	Description
<b><i>Files to Check In</i></b>	This section selects the files you want to check in.
<b><i>Refresh Status</i></b>	Refreshes the current status of items (view, files, and so on) saving you from having to reload the form.  <b>Note:</b> The use of the <i>Show</i> drop-down, to change filtering display, does not automatically refresh status states. The <i>Refresh Status</i> applies any updates without impacting the current filter setting, or check box status.



## Cadence Library Manager User Guide

### Library Manager Forms

Field	Description
<i>Show</i>	<p>A filter pull-down which can change the selection set to display one of the following as current status:</p> <ul style="list-style-type: none"><li>■ <i>Checked Out</i> changes the selection set to list all files that are currently checked out (and <i>by</i> who).</li><li>■ <i>Unmanaged</i> changes the selection set to list all files that are not in the design management workarea</li><li>■ <i>Checked In Writable</i> changes the selection set to list all files that have been checked in and are writable</li><li>■ <i>All</i> restores the original selection set: all files</li></ul> <p>The filters options that are enabled is restricted dependent upon the Check command that is current, mirroring the behavior described for the <i>Check In</i> tab below.</p> <p><b>Note:</b> Click <i>Deselect All</i> to deselect all currently selected files.</p>
<i>Select</i>	Adds files to the selection set.
<i>Deselect All</i>	Remove all files from the selection set.
<i>Invert All</i>	Toggles each checked box to the opposite checked setting.
<b>Comment</b>	<p>This section lets you describe the check-in.</p> <p>When you add the details, such as version information, about a library or cell check-in, the software attaches a copy of the comment to every cellview in the library.</p> <p>You cannot edit or delete the comment after you close the Check In form.</p>
<b>Check In Options</b>	This section displays various check-in options.

## Cadence Library Manager User Guide

### Library Manager Forms

---

Field	Description
	<p>When the <i>Check In</i> command is directly invoked, the Check In form format that is displayed is the same as that for the Check Out form and the Cancel Checkout form. However, the ability to switch command tabs, in the respective forms, is restricted to the command action that is currently applicable, as the legal file sets are mutually exclusive.</p> <p>The only exception is in relation to the <i>Cancel Check Out</i> and <i>Check In</i> tabs, when using the <i>Check Out</i> command, as either of these actions could be applied to checked out files.</p> <p>The <i>Check In</i> command also handles new, or un-managed files, but these file sets would be inappropriate in the context of, for example, a <i>Cancel</i> command.</p>
<i>Use Options</i>	Lets you select the <i>Use Options</i> check box and type any check-in options specific to the particular design management system you are using.
<b>Manage Set</b>	This section reviews and manage the selected files.
<i>Reinitialize</i>	Retrieves the file set that was extracted from the original library manager design management command. All rows being selected.
<i>Refine</i>	Removes any unchecked rows.
<i>OK</i>	<p>Performs the check-in.</p> <p>For certain DM systems, an e-mail message is also sent, to notify team members that a new file is checked in.</p> <p><b>Note:</b> When you check in a library for the first time, you must add the library name to the <code>project.lib</code> (<code>cds.lib</code> may also be used) file before team members can share the data.</p>

---

### ***Related Topics***

[Selecting Items for Copying in the Copy Wizard](#)

[Controlling Automatic Checkin Behavior](#)

## **Cadence Library Manager User Guide**

### **Library Manager Forms**

---

Auto Checkin Environment Variable Settings

Files in Read-Only Mode

## Check Out Form

Use this form to check out a design.

Field	Description
<b><i>Files to Check In</i></b>	This section selects the files you want to check in.
<i>Refresh Status</i>	Refreshes the current status of items (view, files, and so on), saving you from having to reload the form.  <b>Note:</b> The use of the <i>Show</i> drop-down, disabled in the Check Out form, to change filtering display, does not automatically refresh status states. The <i>Refresh Status</i> applies any updates without impacting the current filter setting, or check box status.
<i>Show</i>	A filter pull-down, which is disabled in the Check Out form, as there are no other filter states which can be applied to re-filter and create a different, valid, set.
<i>Select All</i>	Adds all files to the selection set.  For more information about selecting and deselecting items, see <a href="#">Selecting Items for Copying in the Copy Wizard</a> .
<i>Deselect All</i>	Remove all files from the selection set.  <b>Note:</b> To remove a single file from the selection set for check-out, deselect the check box to the left of the file name.
<i>Invert All</i>	Toggles each checked box to the opposite checked setting.
<b><i>Check Out Options</i></b>	This section displays all the check-out options.

## Cadence Library Manager User Guide

### Library Manager Forms

---

Field	Description
	<p>When the <i>Check Out</i> command is directly invoked, the Check Out form format that is displayed, is the same as that for the Check In form and the Cancel Checkout form. However, the ability to switch command tabs, in the respective forms, is restricted to the command action that is currently applicable, as the legal file sets are mutually exclusive.</p> <p>The only exception is in relation to the <i>Cancel Checkout</i> and <i>Check In</i> tabs, when using the <i>Check Out</i> command, as either of these actions could be applied to checked out files.</p> <p>Although all DM systems must return the DM checkout lock, some can perform the checkout again. Additionally, most, if not all, DM systems returns the checked-in version to the workarea. Some DM systems may restore the exact version that you had previously, while others may only return the latest version. That is, some DM systems tracks the older version of workarea before the checkout in order to restore it.</p>
<i>Use Options</i>	<p>Optionally, select the <i>Use Options</i> check box and type any check out options specific to the particular design management system that you want to use.</p>

---

### ***Related Topics***

[Cancel Check Out Form](#)

[Controlling Automatic Checkout Behavior](#)

[Auto Checkout Environment Variable Settings](#)

## Choose Environment File To Load Form

Use this form to load an environment file.

---

Field	Description
<i>Look in</i>	Shows the directory you are searching for an environment file.
<i>File name</i>	Displays the name of the environment file you want to load.
<i>Files of type</i>	Lets you choose one of the following file types to show in the list of files: <ul style="list-style-type: none"><li>■ <i>Cds Environment (*.cdsenv)</i></li><li>■ <i>All Files (*)</i></li></ul>

---

### ***Related Topics***

[Library Manager Environment Variables](#)

## Copy Cell File Form

Use this form to copy a cell file from one library to another library.

Field	Description
<b><i>From</i></b>	This section displays the information of the cell file you want to copy.
<i>Library</i>	Displays the name of the library containing the cell file you want to copy. Defaults to the selected library.
<i>Cell</i>	Displays the name of the cell containing the file you want to copy.
<i>File</i>	Displays the name of the file you want to copy.
<b><i>To</i></b>	This section displays the information of the cell file to where it is copied.
<i>Library</i>	Displays the library to which the cell file is copied.
<i>Cell</i>	Displays the name of the copied cell.
<i>File</i>	Displays the name of the copied file.

### ***Related Topics***

[Copying a Cell File in the Library Manager](#)

## Copy Cell Form

Use this form to copy cellviews to a new cell name.

Field	Description
<b>From</b>	This section displays the information of the cellviews you want to copy.
<i>Library</i>	Displays the name of the library from which you want to copy. Defaults to the selected library.
<i>Cell</i>	Displays the name of the cell you want to copy. Defaults to the selected cell.
<b>To</b>	This section displays the information of the cellviews to where it is copied.
<i>Library</i>	Displays the library to which the library files are copied.
<i>Cell</i>	Displays the new cell name to which the specified cell is copied.
<b>Options</b>	This section lets you control the copy process by specifying limits.
<i>Copy Hierarchical</i>	Copies all referenced cells in your design to the destination library.
<i>Skip Libraries</i>	Prevents cells in the specified reference libraries from being copied.
<i>Edit</i>	Opens a list of libraries that you can select.
<i>Exact hierarchy</i>	Limits the search to the exact hierarchy of your design.
<i>Views To Copy</i>	Copies only the specified views of the selected cell. All Views copies all views of the specified cell. For a hierarchical copy, the Copy command also copies all views of cells instantiated in the specified cell. Select opens a list of view types that you can select from.
<b>Update Instances</b>	This section lets you update the views in the destination cell with the new cell name, when it is set to true.
<i>Of Entire Library</i>	Replaces the existing cell name with the new cell name.
<i>Of New Copies Only</i>	Overwrites only the cellview references you copied from the original library.
<b>Database Integrity</b>	This section updates and validates technology data after the cell is copied.
<i>Re-reference customViaDefs</i>	Re-references <code>customViaDefs</code> to cellviews in the destination library.



## Cadence Library Manager User Guide

### Library Manager Forms

---

---

Field	Description
<i>Check existence in technology database</i>	Validates the existence of layers, purposes, viaDefs, and siteDefs in the technology database of the destination library.
<i>Add To Category</i>	Specifies a new or existing category name to which to add the set of copied cells according to the filter string specified in the <i>Cell</i> field.  By default, the symbol (*) indicates all the copied cells.

---

### ***Related Topics***

[Copying a Cell in the Library Manager](#)

## Copy Cellview Version Form

Use this form to copy cellview versions to a new location.

Field	Description
<b><i>From</i></b>	This section displays the information of the cellview version you want to copy.
<i>Library</i>	Displays the library containing the cellview version to be copied.
<i>Cell</i>	Displays the cell containing the cellview version to be copied.
<i>View</i>	Displays the name of the cellview version to be copied.
<i>Version</i>	Displays the version number of the cellview to be copied.
<b><i>To</i></b>	This section displays the information of the cellview version to where it is copied.
<i>Library</i>	Displays the library in which to copy the cellview version.
<i>Cell</i>	Displays the cell in which to copy the cellview version.
<i>View</i>	Displays the name of the copied cellview version.
<b><i>Copy Options</i></b>	This section allows you to view and select various options for the copied cellview version.
<i>Open After Copy</i>	Opens the copied cellview version for viewing.  For more information, see <a href="#">openView</a> .
<i>Use Options</i>	Selects options specific to your particular design management system.
<i>Defaults</i>	Restores the default values displayed when you initially opened the form.

### ***Related Topics***

[toLibrary](#)

## Copy Library File Form

Use this form to copy a library file to a new location.

---

Field	Description
<b><i>From</i></b>	This section displays the information of the library file you want to copy.
<i>Library</i>	Displays the name of the library containing the file you want to copy. Defaults to the selected library.
<i>File</i>	Displays the name of the file you want to copy.
<b><i>To</i></b>	This section displays the information of the destination library file.
<i>Library</i>	Displays the library to which the library file is copied.
<i>File</i>	Displays the name of the copied file.

---

### ***Related Topics***

[Copy Library Form](#)

## Copy Library Form

Use this form to copy a library to a new location.

Field	Description
<i>From</i>	Displays the name of the library you want to copy. Defaults to the selected library.
<i>To</i>	Displays the library to which the library files are copied.
<b>Options</b>	This section lets you update the copied library.
<i>Update Instances</i>	Update the cells and views in the destination library with the new library name, when it is set to true.
<i>Of Entire Library</i>	Replaces the existing library name with the specified library name.
<i>Of New Copies Only</i>	Overwrites only the cellview references you copied from the original library.
<b>Database Integrity</b>	This section updates and validates technology data after the library is copied.
<i>Re-reference customViaDefs</i>	Re-references <code>customViaDefs</code> to cellviews in the destination library.
<i>Check existence in technology database</i>	Validates the existence of layers, purposes, viaDefs, and siteDefs in the technology database of the destination library.
<i>Apply</i>	Copies all the cellviews to the specified library and leaves the Copy Library form open.

### **Related Topics**

[Copying a Library File](#)

## Copy Preferences Form

Use this form to set preferences for copy and rename operations.

Field	Description
<b>Library and Cell Property Files</b>	This section applies settings to <i>Copy</i> only. You can include properties from library, cell, or both.  For more information, see <a href="#">addLibPropFiles</a> .
<i>Automatically add dependent property files to copy sets</i>	Adds the dependent property files only to the copy sets.  For more information, see <a href="#">addPropFiles</a> and <a href="#">addCellPropFiles</a> .
<i>Do not add dependent property files to copy sets</i>	Excludes the dependent property files to copy sets, which might required by the current design.
<b>Cellview Contents</b>	This section applies settings to <i>Copy</i> only.
<i>Include only the co-managed files of each cellview</i>	Copies only the co-managed files from each cellview.
<i>Include every file inside each cellview</i>	Copies all the available files from each cellview.
<b>Remote Copy Service</b>	This section applies setting to <i>Copy</i> and <i>Rename</i> .
<i>Use session's Copy Service when available</i>	Uses the copy service of the session when it is available.
<i>Use Library Manager Local copy engine only</i>	Uses only the Library Manager local copy engine.
<b>Miscellaneous Settings</b>	This section applies setting to <i>Copy</i> or <i>Rename</i> as indicated.
<i>Enable file progress monitor</i>	Causes the Copy Monitor form to appear during a copy operation.
<i>Warn about Rename of managed data (DM)</i>	Generates a warning whenever any part of the library is renamed, which falls under Design Management. It triggers the Rename Managed Data form to appear.

## Cadence Library Manager User Guide

### Library Manager Forms

---

#### ***Related Topics***

[expandRadio](#)

## Copy View Form

Use this form to copy an existing view to a new location.

Field	Description
<b>From</b>	This section displays the information of the source library
<i>Library</i>	Displays the name of the library from which you want to copy a view. Defaults to the selected library.
<i>Cell</i>	Displays the name of the cell from which you want to copy a view. Defaults to the selected cell.
<i>View</i>	Displays the name of the view you want to copy. Defaults to the selected view.
<b>To</b>	This section displays the information of the destination library
<i>Library</i>	Displays the library to which the specified files are copied.
<i>Cell</i>	Displays the cell to which the specified view is copied.
<i>View</i>	Displays the new view name to which the specified view is copied.
<b>Options</b>	This section lets you manage the copy mechanism in your design.
<i>Copy Hierarchical</i>	Copies all referenced views in your design to the destination cellview.
<i>Skip Libraries</i>	Prevents views in the specified reference libraries from being copied.
<i>Edit</i>	opens a list of libraries that you can select.
<i>Exact hierarchy</i>	Limits the search to the exact hierarchy of your design.
<i>Views To Copy</i>	Copies only the specified views of the selected cell.
<i>All Views</i>	Copies all views of the specified cell. For a hierarchical copy, the Copy command also copies all views of cells instantiated in the specified cell.
<i>Select</i>	Opens a list of view types that you can select from.
<i>Update Instances</i>	Lets you choose one of the following options from the list box.

## Cadence Library Manager User Guide

### Library Manager Forms

---

Field	Description
<i>Of Entire Library</i>	Overwrites instances of the old (library, cell, and view) name with the new name. For example, all instances of .../oldLib/oldCell/oldView are renamed to .../newLib/newCell/newView.
<i>Of New Copies Only</i>	Overwrites only the view references you copied from the original library. For example, only view instances of .../oldCell/oldView are renamed to .../newCell/newView.
<i>Database Integrity</i>	Updates and validates technology data after the view is copied.
<i>Re-reference customViaDefs</i>	Re-references customViaDefs to cellviews in the destination library.
<i>Check existence in technology database</i>	Validates the existence of layers, purposes, viaDefs, and siteDefs in the technology database of the destination library. Warnings, if any, are displayed in the CIW output.
<i>Add To Category</i>	Specifies a new or existing category name to which to add the set of copied cells according to the filter string you type in the <i>Cells</i> field (by default, * to indicate all copied cells).

### ***Related Topics***

[Copying a View in the Library Manager](#)



## Copy Wizard Form (Simple Copy)

Use this form to perform *Simple* copy operation using Copy Wizard.

Field	Description
<i>Generate Copy List</i>	Generates a list of source and destination files for the <i>Hierarchical</i> , <i>By View</i> , and <i>By Configuration</i> copy modes. You do not need to generate a copy list for a <i>Simple</i> copy operation.
<i>Destination Library</i>	Displays the library to which the files are copied.
<b>Update Instances</b>	This section lets you choose one of the following options from the list box:
<i>Of Entire Library</i>	Overwrites instances of the old (library, cell, and view) name with the new name. For example, all instances of <code>.../oldLib/oldCell/oldView</code> are renamed to <code>.../newLib/newCell/newView</code> .
<i>Of New Copies Only</i>	Overwrites only the view references you copied from the original library. For example, only view instances of <code>.../oldCell/oldView</code> are renamed to <code>.../newCell/newView</code> .
<b>Database Integrity</b>	This section updates and validates technology data after the source files are copied.
<i>Re-reference customViaDefs</i>	Re-references customViaDefs to cellviews in the destination library.
<i>Check existence in technology database</i>	Validates the existence of layers, purposes, viaDefs, and siteDefs in the technology database of the destination library. Warnings, if any, are displayed in the CIW output.
<b>Add To Category</b>	This section specifies a new or existing category name to which to add the set of copied cells according to the filter string you type in the <i>Cells</i> field (by default, * to indicate all copied cells).
<i>Select All</i>	Selects all cellviews displayed on the Copy Wizard form.
<i>Deselect All</i>	Deselects all cellviews displayed on the Copy Wizard. The cell names are grayed out, and the check box to the left of the row is deselected. To select an individual row, select the check box.

## Cadence Library Manager User Guide

### Library Manager Forms

---

Field	Description
<i>Clear All</i>	Removes all information from the Copy Wizard. You can type new information in an empty row using one of two methods: <i>Click in the table cell, click the down arrow, and choose from the pop-up menu.</i> <i>Click in the table cell and type your information.</i>
<i>Check Files</i>	Verifies that all the files (cellviews) you specify to be copied are present and have the correct permissions.

---

#### ***Related Topics***

[Performing a Simple Copy Using the Copy Wizard](#)

## Copy Wizard Form (Hierarchical)

Use this form to perform *Hierarchical* copy using Copy Wizard.

Field	Description
<i>Top Library</i>	Displays the name of the library that contains the highest level of the hierarchical design to be copied.
<i>Top Cell</i>	Displays the name of the cell that contains the highest level of the hierarchical design to be copied.
<b><i>Views To Copy</i></b>	This section copies only the specified views of the selected cell.
<i>Names</i>	Specifies the names of the views you want to copy.
<i>Types</i>	Displays the number of types selected from the Select View Types list. The symbol (*) indicates all copied cells.
<i>Select</i>	Opens a list of view types that you can select from.
<i>Skip Libraries</i>	Lets you type the names of libraries that do not contain any elements of the hierarchy and to make the search process more efficient.
<i>Edit</i>	Opens the Skip Libraries Editor dialog box in which you can select the libraries to skip.
<i>Generate Copy List</i>	Changes the Copy List fields to show only the cells and cellviews that are included in the specified hierarchy. The Copy list is empty until you fill in the required information in the <i>Copy Hierarchically</i> group box. It changes the status from <i>not needed</i> to <i>needed</i> whenever you make a change to the Copy List.
	Regenerate the Copy List each time you change information on the Copy Wizard form.
<i>Destination Library</i>	Displays the library to which the files are copied.
<b><i>Update Instances</i></b>	This section lets you choose one of the following options from the list box.

## Cadence Library Manager User Guide

### Library Manager Forms

---

Field	Description
<i>Of Entire Library</i>	Overwrites instances of the old (library, cell, and view) name with the new name. For example, all instances of <code>.../oldLib/oldCell/oldView</code> are renamed to <code>.../newLib/newCell/newView</code> .
<i>Of New Copies Only</i>	Overwrites only the view references you copied from the original library. For example, only view instances of <code>.../oldCell/oldView</code> are renamed to <code>.../newCell/newView</code> .
<i>Database Integrity</i>	Updates and validates technology data after the source files are copied.
<i>Re-reference customViaDefs</i>	Re-references <code>customViaDefs</code> to cellviews in the destination library.
<i>Check existence in technology database</i>	Validates the existence of layers, purposes, viaDefs, and siteDefs in the technology database of the destination library. Warnings, if any, are displayed in the CIW output.
<i>Add To Category</i>	Lets you type a new or existing category name to which to add the set of copied cells according to the filter string you type in the <i>Cells</i> field. By default, the symbol (*) indicates all copied cells.
<i>Select All</i>	Selects all cellviews displayed on the Copy Wizard form.
<i>Deselect All</i>	Deselects all cellviews displayed on the Copy Wizard. The cell names are grayed out, and the check box to the left of the row is deselected. To select an individual row, select the check box.
<i>Clear All</i>	Removes all information from the Copy Wizard. You can type new information in an empty row using one of two methods: <ul style="list-style-type: none"><li>■ By selecting from the various options available in the drop-down list of each cell.</li><li>■ By manually typing information in each cell.</li></ul>
<i>Check Files</i>	Verifies if all the cellviews in the Copy List are available and have the correct permissions.

### ***Related Topics***

#### Copying a Hierarchy Using the Copy Wizard

## Copy Wizard Form (Exact Hierarchy)

Use this form to perform *Exact Hierarchical* copy using Copy Wizard.

Field	Description
<i>Top Library</i>	Displays the name of the library that contains the highest level of the hierarchical design to be copied.
<i>Top Cell</i>	Displays the name of the cell that contains the highest level of the hierarchical design to be copied.
<i>Top View</i>	Displays the name of the view at the highest level of the hierarchical design to be copied.
<i>Extra Views– Names</i>	Lets you type additional view names or expressions to expand the search to include any matching views found in your design hierarchy in the copy operation. If any of these matching views have their own hierarchies, those additional hierarchies are also included.
<i>Select</i>	Opens a list of view types that you can select from.
<i>Skip Libraries</i>	Lets you type the names of libraries that do not contain any elements of the hierarchy and to make the search process more efficient.
<i>Edit</i>	Opens the Skip Libraries Editor dialog box where you can select the libraries to skip.
<i>Generate Copy List</i>	Changes the <i>Copy List</i> fields to show only the cells and cellviews that are included in the specified hierarchy. The Copy List is empty until you fill in the required information in the <i>Copy Hierarchically</i> group box. It changes from <i>not needed</i> to <i>needed</i> whenever you make a change to the Copy List. You must regenerate the Copy List each time you change information on the Copy Wizard form.
<i>Destination Library</i>	Displays the library to which the files are copied.
<i>Update Instances Of Entire Library</i>	<p>Lets you choose one of the following options from the list box:</p> <p>Overwrites instances of the old (library, cell, and view) name with the new name.</p> <p>For example, all instances of .../oldLib/oldCell/oldView are renamed to .../newLib/newCell/newView.</p>

## Cadence Library Manager User Guide

### Library Manager Forms

Field	Description
<i>Of New Copies Only</i>	Overwrites only the view references you copied from the original library.  For example, only view instances of <code>.../oldCell/oldView</code> are renamed to <code>.../newCell/newView</code> .
<b>Database Integrity</b>	This section updates and validates technology data after the hierarchy is copied.
<i>Re-reference customViaDefs</i>	Re-references customViaDefs to cellviews in the destination library.
<i>Check existence in technology database</i>	Validates the existence of layers, purposes, viaDefs, and siteDefs in the technology database of the destination library. Warnings, if any, are displayed in the CIW output.
<i>Add To Category</i>	Lets you type a new or existing category name to which to add the set of copied cells according to the filter string you type in the <i>Cells</i> field. By default, the symbol (*) indicates all copied cells.
<i>Select All</i>	Selects all cellviews displayed on the Copy Wizard form.
<i>Deselect All</i>	Deselects all cellviews displayed on the Copy Wizard. The cell names are grayed out, and the check box to the left of the row is deselected. To select an individual row, select the check box.
<i>Clear All</i>	Removes all information from the Copy Wizard.  You can type new information in an empty row using one of two methods: <ul style="list-style-type: none"><li><input type="checkbox"/> By selecting from the various options available in the drop-down list of each cell.</li><li><input type="checkbox"/> By manually typing information in each cell.</li></ul>
<i>Check Files</i>	Verifies that all the cellviews in the Copy List are present and have the correct permissions.

### **Related Topics**

[Copying an Exact Hierarchy Using the Copy Wizard](#)

## Copy Wizard Form (By View)

Use this form to perform *By View* copy using Copy Wizard.

Field	Description
<i>Library</i>	Lets you type the name of the library from which you want to copy views. You can use only one name.
<i>Cell Filter</i>	Lets you type the criteria for specifying the names of cells whose views you want to copy. You can use an asterisk (*) as a wildcard character. For example, <code>cc*</code> or <code>*a2d</code> . You can specify only one string.
<i>Views To Copy</i>	<p>Specifies the views to copy or criteria for specifying the names of cells whose views you want to copy. You can use an asterisk (*) as a wildcard character.</p> <p>For example, to specify only one string, specify <code>sym*</code>.</p>
<i>Select</i>	Opens a list of view types that you can select from.
<i>Generate Copy List</i>	Changes the copy list to show only the specified cells and cellviews. None are displayed until you fill in the required information in the <i>Copy By View</i> group box. It changes from <i>not needed</i> to <i>needed</i> whenever you make a change to the copy list. You must regenerate the copy list each time you change information on the Copy Wizard form.
<i>Destination Library</i>	Displays the library to which the files are copied.
<b>Update Instances</b>	This section lets you choose one of the following options from the list box:
<i>Of Entire Library</i>	<p>Overwrites instances of the old (library, cell, and view) name with the new name.</p> <p>For example, all instances of <code>.../oldLib/oldCell/oldView</code> are renamed to <code>.../newLib/newCell/newView</code>.</p>
<i>Of New Copies Only</i>	Overwrites only the view references you copied from the original library. For example, only view instances of <code>.../oldCell/oldView</code> are renamed to <code>.../newCell/newView</code> .
<b>Database Integrity</b>	This section updates and validates technology data after the source files are copied.
<i>Re-reference customViaDefs</i>	Re-references <code>customViaDefs</code> to cellviews in the destination library.

## Cadence Library Manager User Guide

### Library Manager Forms

---

Field	Description
<i>Check existence in technology database</i>	Validates the existence of layers, purposes, viaDefs, and siteDefs in the technology database of the destination library. Warnings, if any, are displayed in the CIW output.
<i>Add To Category</i>	Lets you type a new or existing category name to which to add the set of copied cells according to the filter string you type in the <i>Cells</i> field. By default, the symbol (*) indicates all copied cells.
<i>Select All</i>	Selects all cellviews displayed on the Copy Wizard form.
<i>Deselect All</i>	Deselects all cellviews displayed on the Copy Wizard form. The cell names are grayed out, and the check box to the left of the row is deselected. To select an individual row, select the check box.
<i>Clear All</i>	<p>Removes all information from the Copy Wizard form.</p> <p>You can type new information in an empty row using one of two methods:</p> <ul style="list-style-type: none"><li>■ By selecting from the various options available in the drop-down list of each cell.</li><li>■ By manually typing information in each cell.</li></ul>
<i>Check Files</i>	Verifies that all the cellviews in the copy list are present and have the correct permissions.

### ***Related Topics***

[Copying Specific View Using the Copy Wizard](#)



## Copy Wizard Form (By Configuration)

Use this form to perform *By Configuration* copy using Copy Wizard.

Field	Description
<i>Library</i>	Lets you type the name of the library from which you want to copy the configuration view. You can use only one name.
<i>Cell</i>	Lets you type the cell name of the configuration to be copied.
<i>Config View</i>	Lets you specify the name of the configuration to copy.
<i>Skip Libraries</i>	Lets you type the names of libraries that do not contain any elements of the configuration (to make the search process more efficient).
<i>Edit</i>	Opens the Skip Libraries Editor dialog box in which you can select the libraries to skip.
<i>Generate Copy List</i>	Changes the list to show only the cells and cellviews that are included in the specified configuration. None are displayed until you fill in the required information in the fields above. It changes from <i>not needed</i> to <i>needed</i> whenever you make a change to the copy list. You must regenerate the copy list each time you change information on the Copy Wizard form.
<i>Destination Library</i>	Displays the library to which the files are copied.
<b>Update Instances</b>	This section lets you choose one of the following options from the list box:
<i>Of Entire Library</i>	Overwrites instances of the old (library, cell, and view) name with the new name.  For example, all instances of <code>.../oldLib/oldCell/oldView</code> are renamed to <code>.../newLib/newCell/newView</code> .
<i>Of New Copies Only</i>	Overwrites only the view references you copied from the original library. For example, only view instances of <code>.../oldCell/oldView</code> are renamed to <code>.../newCell/newView</code> .
<b>Database Integrity</b>	This section updates and validates technology data after the configuration is copied.
<i>Re-reference customViaDefs</i>	Re-references <code>customViaDefs</code> to cellviews in the destination library.

## Cadence Library Manager User Guide

### Library Manager Forms

---

Field	Description
<i>Check existence in technology database</i>	Validates the existence of layers, purposes, viaDefs, and siteDefs in the technology database of the destination library. Warnings, if any, are displayed in the CIW output.
<i>Add To Category</i>	Lets you type a new or existing category name to which to add the set of copied cells according to the filter string you type in the <i>Cells</i> field. By default, the symbol (*) indicates all copied cells.
<i>Select All</i>	Selects all cellviews displayed on the Copy Wizard.
<i>Deselect All</i>	Deselects all cellviews displayed on the Copy Wizard. The cell names are grayed out and the check boxes to the left of the row are deselected. To select an individual row, select the check box.
<i>Clear All</i>	Removes all information from the Copy Wizard.  You can type new information in an empty row using one of two methods: <ul style="list-style-type: none"><li>■ By selecting from the various options available in the drop-down list of each cell.</li><li>■ By manually typing information in each cell.</li></ul>
<i>Check Files</i>	Verifies that all the cellviews in the copy list are present and have the correct permissions.

### ***Related Topics***

[Copying Specific Cells in a Configuration Using the Copy Wizard](#)

## Delete By View Form

Use this form to delete views from cells

Field	Description
<i>Library Name</i>	Displays the library of the selected view.
<i>Cell Filter</i>	Lets you type the criteria for specifying the names of cells whose views you want to delete. You can use an asterisk (*) as a wildcard character, such as <code>cc*</code> or <code>*a2d</code> . You can specify only one string.
<i>View Filter</i>	Lets you specify the view you want to delete from the specified cell or cells, in the selected library. Click the down arrow to display a list box from which you can select a view name.
<i>Add View Name To Selection List</i>	Lets you add a view name to the <i>View Filter</i> list box.
<i>Find Copied Versions</i>	Lets you delete only cellview versions you have previously copied.

### ***Related Topics***

[Deleting Cells Using Filters](#)

## Delete Cells Form

Use this form to delete cells from the design.

Field	Description
<i>Delete</i>	Lists all the cells in the design that you want to delete.
<i>Don't Delete</i>	Lists all the cells in the design that you do not want to delete.
<i>Select</i>	Lets you specify a filter string for selecting all matching cell names in a list box.
<b>Options</b>	This section lets you choose to delete items locally only or also from the design management repository.
<i>Delete Local And Inactive From DM System</i>	Deletes your local copy and the copy in the design management repository.
<i>Delete Local Only</i>	Deletes the specified items from your Library Manager but not from the design management repository.

### ***Related Topics***

[Deleting a Cell](#)

## Delete Cell Views Form

Use this form to delete cellviews from the design.

Field	Description
<i>Delete</i>	Lists all the cellviews in the design that you want to delete.
<i>Don't Delete</i>	Lists all the cellviews in the design that you do not want to delete.
<i>Select</i>	Lets you specify a filter string for selecting all matching cellview names in a list box.
<b>Options</b>	This section lets you choose to delete items locally only or also from the design management repository.
<i>Delete Local And Inactive From DM System</i>	Deletes your local copy and the copy in the design management repository.
<i>Delete Local Only</i>	Deletes the specified items from your Library Manager but not from the design management repository.

### **Related Topics**

[Deleting a View](#)

## Delete Libraries Form

Use this form to delete libraries from the design.

Field	Description
<i>Delete</i>	Lists all the libraries in the design that you want to delete.
<i>Don't Delete</i>	Lists all the libraries in the design that you do not want to delete.
<i>Select</i>	Lets you specify a filter string for selecting all matching library names in a list box.
<b>Options</b>	This section lets you choose to delete items locally only or also from the design management repository.
<i>Delete Local And Inactive From DM System</i>	Deletes your local copy and the copy in the design management repository.
<i>Delete Local Only</i>	Deletes the specified items from your Library Manager but not from the design management repository.

### ***Related Topics***

[Deleting a Library](#)

[Selecting and Moving Data in the Library Manager](#)

## Delete Library Views Form

The Delete Library Views form displays a list of the paths for views you chose to delete on the Delete By View form. The name of the library containing these views is displayed above the list.

- Views with a selected check box to the left of the row are selected for deletion.
- Views with a deselected check box does not get deleted.

You can select or deselect views by clicking the toggle button to the left of the row.

---

Field	Description
<i>Select All</i>	Selects all views and selects the check box to the left of each row.
<i>Deselect All</i>	Deselects all views and selects all check boxes.
<b>Options</b>	This section lets you choose to delete items locally only or also from the design management repository.
<i>Delete Local And Inactive From DM System</i>	Deletes your local copy and the copy in the design management repository.
<i>Delete Local Only</i>	Deletes the specified items from your Library Manager but not from the design management repository.

---

### ***Related Topics***

[Deleting Cells Using Filters](#)

## Display Options Form

Use this form to modify the UI of a library.

Field	Description
<b><i>For Objects</i></b>	This section contains two tabs, one tab for <i>Library</i> specific settings (for example, overrides and icons) and another tab for <i>View</i> specific settings.  The only option currently in the latter tab is for <i>Show extended states</i> , which turns on or off DM queries, and other information that is displayed in the tables of the Library Manager window - views and files.  Switching this option off restores the display behavior.
<i>Display Overrides</i>	Lets you override custom display settings for libraries.
<i>Show all libraries using standard style</i>	Overrides any custom display settings and displays all libraries in the default style.
<i>Show hidden libraries</i>	Overrides any invisible settings on libraries and displays all libraries.
<i>Show library colors</i>	Displays libraries in the customized colors.
<i>Show custom library icons</i>	Displays custom icons for libraries. Icons are visible in both the Tree view and Lists view.
<i>Show Lists view library icons</i>	Displays custom icons only in the Lists view of libraries. Icons are not displayed in the Tree view.
<b><i>State Analysis</i></b>	This section allows you to set the Design Management states.



## Cadence Library Manager User Guide

### Library Manager Forms

Field	Description
<i>Enable query of Design Management states</i>	<p>Deselects this option if your Design Management performance is slow and impacting library browsing speed. Consequently, Design Management icons does not get displayed and related Design Management state is shown as empty in the Library Manager extended tables.</p> <p>This option and <i>Show extended states</i> must both be set to on in order to see DM states in the main Library Manager window.</p> <p>The <i>Enable delay before DM syncs</i> option enables you in quicker browsing when slower DM integrations for GDM are in use and you do not want to turn off the DM display feature entirely. This option is enabled once you select the <i>Enable query of data management</i> check box.</p> <p>The DM gets queried on a selected item, such as a cell, including its views and any other files, are displayed <i>n</i> seconds after the current cell is selected in the list. The select action loads the item's contents, which includes views and files of the specified cells into the list display.</p> <p>The value in the <i>Wait</i> field (in seconds) comes from the <code>.cdsenv</code> DB setting. By default, the value is 2.5 seconds. The accepted values are 0 through 999 seconds. Set the wait time value higher if you know that your DM system gets slow in responding to such queries. Once your selection/browsing has stopped, and the Wait delay time-out is reached, the DM system is then queried to update the DM status display in the window for the current library or cell contents.</p>
<i>Enable delay before DM syncs</i>	<p>Enables a delay in the time period between when a cell was selected and when the Design Management status requests gets called for it. A delay enables you to quickly select between several different cells in a library without the updated DM status being obtained for all of those cells. You can set the delay period by specifying the units in seconds in the <i>Wait ...seconds after selecting</i> field.</p> <p>By default, the <i>Enable delay before Design Management syncs</i> check box is not selected.</p>
<i>Enable poll</i>	<p>Reanalyzes states periodically, based on the two slider and spin-box value settings for the related options, as follows:</p>

## Cadence Library Manager User Guide

### Library Manager Forms

---

Field	Description
<i>Interval</i>	Sets the number of seconds to determine the time setting between successive reanalyze state queries.
<i>Idle limit over interval</i>	<p>Determines how long the polling should continue after the last UI interaction. The seconds value set here is added to the interval seconds set.</p> <p>The polling features enable state change detections without external notification channels, such as Design M check out from another system, workarea, and/or user. Some common edit operations in Virtuoso also notifies the Library Manager. In such cases however, the polling settings are not used to update values.</p> <p>Design Management systems that support large number of clients should advise users to select longer poll intervals in order to prevent overwhelming shared resources. The fastest interval value is 1 second, while the default is 30 seconds. These poll settings can be obtained by referring to the <code>.cdsenv</code> file. Virtuoso edit notifications occur independently of poll settings. Fast poll intervals are therefore not necessary here and may impact all user's DM operations.</p>
<b>Resources</b>	This section includes an option to customize the display settings.
<i>Custom library display attributes</i>	Opens the Display Settings form, which lets you view, edit, or create library display settings.

### **Related Topics**

[Overriding Customized Library Display Settings](#)

[Display Settings Form](#)

## Display Settings Form

Use this form to customize display settings for the libraries.

Field	Description
<b><i>Library Display Attributes</i></b>	This section lists the attributes, both predefined and custom, that you can set on libraries.
<i>Used by libraries</i>	Specifies the number of libraries on which the selected attribute is set.
<i>Add</i>	Opens the Add Display Attribute form, which lets you create new attributes.
<b><i>Display Libraries</i></b>	This section displays the currently-selected display options for the selected attribute and lets you modify them.
<i>As standard</i>	Displays libraries in the default style.
<i>As hidden</i>	Makes libraries invisible.
<i>Using color</i>	Displays libraries in the specified color. The <i>Select</i> button opens the Select color dialog box from which you can choose a color.
<i>Using icon</i>	Displays the specified icon next to libraries. The <i>Select</i> button opens the Select icon dialog box from which you can choose an icon.
<i>Allow dynamic overrides</i>	Applies overrides to libraries using the display attribute.
<b><i>Preview</i></b>	Displays a sample library name with the specified display options.

### ***Related Topic***

[Library Display Settings](#)

## DM File Status Form

Use this form to view the design management status of all cells in a library, all views for a cell, and individual files.

Field	Description
<b>File Status</b>	This section lets you select the files to view the DM status.
<i>Refresh Status</i>	Refreshes the current status of items and prevent saving the form.  <b>Note:</b> The use of the <i>Show</i> drop-down, to change filtering display, does not automatically refresh status states. The <i>Refresh Status</i> applies any updates without impacting the current filter setting, or check box status.
<i>Show</i>	A filter pull-down, which is only enabled if the <i>Status</i> tab is current in the DM File Status form. Here, you can choose to filter and refine the content of the <i>Status Command</i> section.  <b>Note:</b> This option is not enabled for other DM commands, such as <i>Check In</i> and <i>Check Out</i> , as their initial input sets are pre-defined by one or more specific states which only apply to that command.
<i>Select All</i>	Adds all files to the selection set.
<i>Deselect All</i>	Remove all files from the selection set.
<i>Invert All</i>	Toggles each checked box to the opposite checked setting.
<b>Manage Set</b>	This section lets you review and manage the selected files.
<i>Reinitialize</i>	Retrieves the file set that was extracted from the original library manager design management command. All rows being selected.
<i>Refine</i>	Removes any unchecked rows.

### ***Related Topic***

[Design Management File Status](#)

# **Cadence Library Manager User Guide**

## **Library Manager Forms**

---

## New Category Form

Use this form to edit the categories.

Field	Description
<i>Category Name</i>	Displays the name of the category to be edited.
<b>Cells</b>	This section lists all the selected and unselected cells.
<i>Not In Category</i>	Lists all the cells in the design that are not in the selected category.
<i>In Category</i>	Lists all the cells in the design that are in the selected category.
<b>Sub Categories</b>	This section lists all the selected and unselected subcategories.
<i>Not In Category</i>	Lists all the subcategories in the library that are not in the category.
<i>In Category</i>	Lists all the subcategories that are in the category.

### ***Related Topics***

[Editing a Category Using Library Manager](#)

## Library Browser Form

This form saves the information you type and restores it each time you reopen the form or the Virtuoso session.

---

Field	Description
<i>Show Categories</i>	Expands the form to display categories.
<i>Show Non-Virtuoso View Types</i>	Displays non-Virtuoso view types for the selected cell.
<i>Library</i>	Displays the names of the libraries specified in your <code>cds.lib</code> file.
<i>Cell</i>	Displays the names of the cells in the selected library.
<i>View</i>	Displays the names of the views in the selected cell.

---

### ***Related Topics***

[Opening the Library Browser Form](#)

[showNonVirtuosoViewtypes](#)

[Library Manager Customization](#)

## Library Manager Form

Use this form to manage the libraries of a design.

Field	Description
<i>File</i>	Lets you open or create libraries, cellviews, or categories or open a UNIX shell window.
<i>Edit</i>	Edits libraries, cellviews, categories, or your <code>cds.lib</code> file.
<i>View</i>	Lets you specify filters for displaying cellviews.
<i>Design Manager</i>	Lets you access design management functions.  <b>Note:</b> Commands on the <i>Design Manager</i> menu are active only if you are using a design management system.
<i>Show Categories</i>	Displays the <i>Category</i> list box.
<i>Show Files</i>	Displays the <i>Files in</i> list box.
<i>Library</i>	Displays the libraries defined in your <code>cds.lib</code> file.
<i>Category</i>	Displays the categories in the selected library.
<i>Cell</i>	Displays the cells in the selected library or selected category.
<i>View</i>	Displays the views in the selected cell.
<i>Files in</i>	Expands the form to list the available files. <ul style="list-style-type: none"><li>■ <i>Library</i>: Lists the files available in the selected library.</li><li>■ <i>Cell</i>: Lists the files available in the selected cell.</li></ul>
<i>Messages</i>	Displays information about actions initiated from Library Manager.

### ***Related Topics***

[View Property Editor Form](#)



## Library Property Editor Form

Use this form to edit the properties of a library.

Field	Description
<b>Library</b>	This section displays the information of the selected library.
<i>Name</i>	Displays the view name of the selected library.
<i>Owner</i>	Displays the user ID of the owner of the selected library.
<i>Group</i>	Displays the group of the owner of the selected library.
<i>Last Modify</i>	Displays the date/time stamp of the last modification to the library.
<i>Read Path</i>	Displays the path to a read-only version of the library.
<i>Write Path</i>	You cannot change any of the values for the library.
<b>UNIX Permissions Mode</b>	This section displays the read/write/execute permissions for the owner, group, and other.  You cannot change any of these permissions.
<i>Owner</i>	Displays the read/write/execute permissions for the owner of the library.
<i>Group</i>	Displays the read/write/execute permissions for the group of the owner of the library.
<i>Other</i>	Displays the read/write/execute permissions for anyone who has access to the library.
<b>Properties</b>	Lists all the properties available in the specified library at the bottom of the Library Property Editor form and you can edit them on this form.  If the specified library has no properties then <i>No Property Attached</i> text is displayed.
<i>Add</i>	Adds a property by opening the Add Property form.
<i>Delete</i>	Removes the selected property.
<i>Modify</i>	Opens the Modify ' <i>propertyName</i> ' form.

## Cadence Library Manager User Guide

### Library Manager Forms

---

#### ***Related Topics***

[Editing Library Properties](#)

[Add Property Form](#)

[Modify '\*propertyName\*' form](#)

## Load Technology File Form

Use this form to associate a new library with a technology file.

---

Field	Description
<i>ASCII Technology File</i>	Lets you type the path to a technology file to associate with a new library. Alternatively, click the Browse button to locate the technology file.
<i>New Technology Library</i>	Displays the name of the new library that gets associated with the technology file.

---

### ***Related Topics***

[Compiling an ASCII Technology File](#)

## Modify ‘propertyName’ Form

Use this form to modify the property values.

---

Field	Description
<i>Defaults</i>	Clears all values in the fields and sets <i>Type</i> to <i>int</i> .
<i>Name</i>	Displays the name of the property.
<i>Type</i>	Lets you change the property type. Depending on the type you choose, the remaining fields change to prompt you for values for the property.
<i>Value</i>	Lets you change the default value for the property.
<i>Minimum Value</i>	Lets you change the minimum value for the property.
<i>Maximum Value</i>	Lets you change the maximum value for the property.
	Used for <i>int</i> , <i>float</i> , and <i>time</i> property types. The View Property Editor form or Cell Property Editor form displays these minimum and maximum values next to the property name.
<i>Possible Choices</i>	Lets you change the possible property values. Used for <i>string</i> property type only. The View Property Editor form or the Cell Property Editor form creates a list box of these choices.

---

### ***Related Topics***

[Library Property Editor Form](#)

[Modifying Properties of a Library, Cell, or View](#)

## New Category Form

Use this form to create a new category.

Field	Description
<i>Category Name</i>	Lets you type the name of the new category you want to create.
<b>Cells</b>	This section lists all the cells available in the design.
<i>Not In Category</i>	Lists all the cells in the design that are not in the new category.
<i>In Category</i>	Lists all the cells in the design that are in the new category.
<b>Sub Categories</b>	This section lists all the subcategories available in the design.
<i>Not In Category</i>	Lists all the subcategories in the library that are not in the new category.
<i>In Category</i>	Lists all the subcategories in the library that are in the new category.

### ***Related Topics***

[Creating a Category Using Library Manager](#)

## New File Form

Use this form to create a new cellview.

Field	Description
<i>Library</i>	Lets you choose the library in which you create the new cellview.
<i>Cell</i>	Displays the name of the new cell.
<i>View</i>	Displays the name of the new view.  For the <i>Cell</i> and <i>View</i> name fields, only legal identifiers in the CDBA name space can be used. For example, <i>white spaces</i> cannot be used in a field name.
<i>Type</i>	Displays the type of cellview of the selected <i>View</i> .
<b>Application</b>	This section lets you choose the application you want to work using the specified type of cellview.
<i>Open with</i>	Lets you choose which application should be opened to work with a particular <i>Type</i> of file.
<i>Always use this application for this type of file</i>	Sets the application currently selected in the <i>Open with</i> option to always be used when the current view <i>Type</i> is chosen.
<i>Library path file</i>	Displays the path to the <code>cds.lib</code> file you are using. You cannot edit this field.

### **Related Topics**

[Creating a New Cellview](#)

## New Library Form

Use this form to create a new library.

---

Field	Description
<b><i>Library</i></b>	This section lets you specify a new library name and path.
<i>Name</i>	Lets you specify a name for the new library.
<i>Directory</i>	Lists the directories in the path displayed below the list box. You can select a directory from this list or specify a path to a directory in the field below the list.
<b><i>Design Manager</i></b>	This section lets you choose your design management setup. A list box gets displayed with the options if more than one exists.
<i>Use None</i>	Indicates that there is no design management system currently available to use.
<i>Use No DM</i>	Indicates that you do not want the library to be placed under design management control, whether there is a design management system available there or not, now or later. If there is one available, you can still decide to check it in later on.
<i>Compression enabled</i>	Enables you write OpenAccess data to the library in a compressed format.

---

### ***Related Topics***

[Creating a New Library in the Library Manager](#)

## Reference Existing Technology Libraries Form

Use this form to set the reference technology libraries for the new library.

Field	Description
<i>New Library</i>	Displays the new library that is to use the reference technology libraries.
<i>Technology Libraries</i>	Lists the available libraries to be referenced.
<b>Reference Technology Libraries</b>	This section lists the existing libraries which is used as a reference by the new library for technology information.
<i>Up</i>	Moves the selected referenced technology library up the reference priority.
<i>Down</i>	Moves the selected referenced technology library down the reference priority.

### ***Related Topics***

[Selecting and Moving Data in the Library Manager](#)



## Rename Cell Form

Use this form to change the name of a cell.

---

Field	Description
<i>From Cell</i>	Displays the original cell name.
<i>To Cell</i>	Displays the new cell name that you type.
<i>Update Instances</i>	Updates all instances that reference the old name with the new name.
<i>Design Management Options</i>	<p>Appears only when a DM system is being used. The options appear enabled only when the DM system supports renaming. You can specify the following options:</p> <p>The process stops if errors are found in the precheck phase. You can specify whether it should stop after the first error is found or after all errors are found.</p> <p>In <i>Check In Options</i>, you may specify comments in the <i>Comment</i> box or specify an options file in the <i>Use Options</i> field.</p> <p><b>Note:</b> If you are using a design management system that does not support GDM rename, the Library Manager renames design-managed objects with only the version available for its cellviews and files.</p>

---

### ***Related Topics***

[Selecting and Moving Data in the Library Manager](#)

[Renaming a Library](#)

## Rename File Form

Use this form to change the name of a file.

---

Field	Description
<i>From File</i>	Displays the original file name.
<i>To File</i>	Displays the new file name that you type.
<i>Update Instances</i>	Updates all instances that reference the old name with the new name.
<i>Design Management Options</i>	<p>Appears only when a DM system is being used. The options appear enabled only when the DM system supports renaming. You can specify the following options:</p> <p>The process stops if errors are found in the precheck phase. You can specify whether it should stop after the first error is found or after all errors are found.</p> <p>In <i>Check In Options</i>, you may specify comments in the <i>Comment</i> box or specify an options file in the <i>Use Options</i> field.</p>

---

### ***Related Topics***

[Renaming Files](#)

## Rename Library Form

Use this form to change the name of a library.

---

Field	Description
<i>From Library</i>	Displays the original library name.
<i>To Library</i>	Displays the new library name that you type.
<i>Update Instances</i>	Updates all instances that reference the old name with the new name.
<i>Design Management Options</i>	<p>Appears only when a DM system is being used. The options appear enabled only when the DM system supports renaming. You can specify the following options:</p> <p>The process stops if errors are found in the precheck phase. You can specify whether it should stop after the first error is found or after all errors are found.</p> <p>In <i>Check In Options</i>, you may specify comments in the <i>Comment</i> box or specify an options file in the <i>Use Options</i> field.</p>

---

### ***Related Topics***

[Renaming a Library](#)

## Rename Reference Library Form

Use this form to change the name of a reference library.

---

Field	Description
<i>In Library</i>	Displays the name of the design library that uses a reference library whose name you want to change.
<i>From Library</i>	Displays the name of the current reference library.
<i>To Library</i>	Displays the name of the new reference library.
<i>Refresh Session</i>	Refreshes the session.

---

### ***Related Topics***

[Selecting and Moving Data in the Library Manager](#)

## Rename View Form

Use this form to change the view name of a cell.

---

Field	Description
<i>From View</i>	Displays the original view name.
<i>To View</i>	Displays the new view name that you specified.
<i>Update Instances</i>	Updates all instances that reference the old name with the new name.
<i>Design Management Options</i>	<p>Appears only when a Design Management system is being used. The options appear enabled only when the Design Management system supports renaming.</p> <p>You can specify the following options:</p> <ul style="list-style-type: none"><li><input type="checkbox"/> The process stops if errors are found in the precheck phase. You can specify whether it should stop after the first error is found or after all errors are found.</li><li><input type="checkbox"/> In <i>Check In Options</i>, you may specify comments in the <i>Comment</i> box or specify an options file in the <i>Use Options</i> field.</li></ul>

---

### ***Related Topics***

[Renaming a Library](#)

## Save Library Manager Defaults Form

Use this form to save the default file.

Field	Description
<i>Save Defaults File As</i>	Displays the file name you type as the new file name.
<b>Filter</b>	This section displays the path and file name you type, including wildcard characters, to point to the original file from where you want to save.
<i>Filter</i>	Uses the path you typed in the <i>Filter</i> field as the filter for your directory search lists.
<b>Directory</b>	This section shows the directories available at the end of the path.
<i>Files of type</i>	Shows the files available at the end of the path.
<b>Options</b>	This section lets you specify save options.
<i>Save</i>	Saves all the possible values to your <code>.cdsenv</code> file in the directory you specified, if you select the <i>All possible values</i> check box from the <i>Save</i> option.
<i>File Status</i>	In this section, you need to specify the values to want to save.
<i>Overwrite</i>	Saves the values you type by overwriting your <code>.cdsenv</code> file.
<i>Merge values</i>	Saves the values you modify into your <code>.cdsenv</code> file. However, it does not delete the pre-existing unmodified values.
<i>Retain values</i>	Saves the values you specify by creating another file. You must type a different file name in the <i>Save Defaults File As</i> field.

### Related Topics

[Saving Settings to a `.cdsenv` File](#)

## Select an icon Form

Use this form to add icons to the attribute.

Field	Description
<b>Icon Source</b>	This section lets you select the directories from which you want to display icon files in the <i>Files</i> list.
<i>All directories in Cadence search path (setup.loc)</i>	<p>Displays icons from all icon directories found by CSF search in the <i>Files</i> list. Specifically, icons from the following sub-directories of every location specified in your <code>setup.loc</code> file are displayed:</p> <pre>icons/library/16x16</pre> <pre>icons/16x16</pre> <p>If an icon definition is found in multiple locations, the definition from the location that has higher precedence in the <code>setup.loc</code> file is used. If an icon is found in both an <code>icons/library/16x16</code> and an <code>icons/16x16</code> directory, the icon from the <code>icons/library/16x16</code> directory is used.</p> <p>You can place the cursor over an icon file name in the <i>Files</i> list to see which directory it is obtained from.</p>
<i>Specific directory</i>	<p>Lets you select a specific directory from which to display icon files.</p> <p>All directories found by CSF search that contain an <code>icons/library/16x16</code> or an <code>icons/16x16</code> subdirectory are listed in the drop-down list. Until you create your custom icon directories, only the directories containing Cadence application icons gets listed in this field, such as the <code>your_install_dir/share/cdssetup/icons/16x16</code> icon directory.</p>
<b>Files</b>	This section displays all icon files found in the directories specified in <i>Icon Source</i> .
<i>Show names containing</i>	Lets you specify a pattern to filter icon file names. Only the file names containing the pattern are displayed.

***Related topics***

Selecting an Icon for a Library Display Attribute



## Submit Form

Use this form to submit files to the project design management repository.

Field	Description
<i>Files to Submit</i>	Lets you select files to submit to the project design management repository.
<i>Select All</i>	Selects all files in the list for submitting.
<i>Deselect All</i>	Deselects all files in the list so that no files are submitted.
<i>Description (Optional)</i>	Lets you type a description of your design changes.
<b><i>Submit Options</i></b>	This section lets you modify the submit operation.
<i>Request Name</i>	Lets you assign a name to the integration request.
<i>Use Options</i>	Lets you specify the options you want to use for the submit operation.

### ***Related Topics***

[Submitting Changes to the Design Management System](#)

## Technology File for New Library Form

Use this form to attach or create a technology file to a library.

---

Field	Description
<i>Compile an ASCII technology file</i>	Attaches a specified technology file to the new library.
<i>Reference existing technology libraries</i>	Create a new technology file and reference it to an existing technology library.
<i>Attach to an existing technology library</i>	Opens a form that lets you select a technology file from the list for the new library.
<i>Do not need process information</i>	Lets you create a new library that does not need a technology file. By default, the system attaches the <code>default.tf</code> file the first time you access the library from a Virtuoso product.)

---

### ***Related Topics***

[Creating a New Library in the Library Manager](#)

[Compiling an ASCII Technology File](#)

[Referencing Existing Technology Libraries](#)

## Version Information Form

Use this form to update and check the version information of files.

Field	Description
<i>Check Out</i>	Checks out the project default version of a file or cellview, which is already checked in to your workarea. The Version Information form shows that the file is checked out.  You need to select the version from the list box for a file that has been checked in and then the selected version is checked out.
<i>Check In</i>	Checks in a file or cellview. Select the version from the list box to be checked in and the selected version is checked in.
<i>Cancel Check Out</i>	Cancels the check out of a file or cellview. Select the version from the list box to have its checked out canceled.  The selected version check out is canceled.
<i>Update</i>	Lets you specify which version, other than the project default, is read in to your workarea when you open a file in read-only mode.  The software updates the workarea with the specified version.
<i>Rollback</i>	Lets you roll back a version so it becomes the project default version and updates the Version Information form to reflect the rollback
<i>Copy</i>	Lets you copy versions of a cellview that has been checked in.  These commands are active only for checked-in files. You can use the Version Information form to display version information about checked-out files, but these commands are inactive.
<i>Description</i>	Displays the description associated with the action performed.
<i>Refresh History</i>	Performs another query of the DM system for the current history and status, and then updates the form displayed.  <b>Note:</b> This is the same behavior as when a new cellview is selected in the Library Manager main window.

### ***Related Topics***

[Cellview and File Versions](#)

## View Property Editor Form

Use this form to edit the properties of the selected views.

Field	Description
<i>Add</i>	Opens the Add Property form.
<i>Delete</i>	Removes the selected property.
<i>Modify</i>	Opens the Modify <i>propertyName</i> form.
<b>View</b>	This section displays the information of the selected cellview.
<i>Name</i>	Displays the view name of the selected cellview.
<i>Owner</i>	Displays the user ID of the owner of the selected cellview.
<i>Group</i>	Displays the group of the owner of the selected cellview.
<i>Last Modify</i>	Displays the date and time of the last modification to the cellview.
<i>Read Path</i>	Displays the path to a read-only version of the cellview.
<i>Write Path</i>	Displays the path to a writable version of the cellview.
<b>UNIX Permissions Mode</b>	This section displays the read/write/execute permissions for the owner, group, and other.  You cannot change any of these permissions.
<i>Owner</i>	Displays the read/write/execute permissions for the owner of the cellview.
<i>Group</i>	Displays the read/write/execute permissions for the group of the owner of the cellview.
<i>Other</i>	Displays the read/write/execute permissions for anyone who has access to the cellview.
<b>Properties</b>	Lists all the properties available in the specified view at the bottom of the View Property Editor form and you can edit them on this form.  If the specified view has no properties then <i>No Property Attached</i> text is displayed.

## Cadence Library Manager User Guide

### Library Manager Forms

---

#### ***Related Topics***

[Cellview and File Versions](#)

[Editing View Properties](#)