ModelLib

Simucad Model Library Installation and Configuration

SIMUCAD Design Automation, Inc.
4701 Patrick Henry Drive, Bldg. 2
Santa Clara, CA 95054
Telephone (408) 567-1000
Internet http://www.simucad.com

March 6, 2007

ModelLib

Copyright 2007

SIMUCAD Design Automation, Inc
4701 Patrick Henry Drive, Building #2
Santa Clara, CA 95054

Phone: (408) 567-1000
Internet: http://www.simucad.com

Simucad Model Library Installation and Configuration

Introduction

The Simucad Model Library (MODELLIB) is a collection of independent SPICE Models that are loaded when a MODELLIB based circuit simulator runs. Using MODELLIB, customers can enjoy a fast turnaround for SPICE model updates. MODELLIB also enables rapid collaborative development and exchange of custom SPICE models between academic and industry partners. Customers using SMARTSPICE, SMARTSPICERF, HARMONY and TWISTER can download updated SPICE models in MODELLIB form from the Resource Center section of Simucad's website. This document is a detailed installation procedure for administrators and users of MODELLIB. Developers of SPICE models for use in MODELLIB based simulators, please refer to the SIMUCAD MODEL LIBRARY DEVELOPERS GUIDE.

MODELLIB enables the continued use of qualified versions of core simulators while allowing updates to individual models. New models can be rolled out either site wide or to individual users. This enables rapid development and updates of SPICE models without effecting the stability and reliability of the core simulation engine.

MODELLIB enables on site compilation and development of custom SPICE models that can be used within MODELLIB based simulations. Simucad Model Library Development Kits are available for several models. These development kits serve as examples of how to implement custom models within MODELLIB based applications.

Overview

A MODELLIB based simulator loads a set of SPICE models from the Simucad Model Library when it is run. The list of libraries loaded at runtime are defined in configuration files that inform the simulator of the path to the individual model libraries. Modifying and updating the MODELLIB Configurations is performed by using only the `slconfig` command or the management console (`sman`). Do not hand edit the file.

The factory default configuration is stored with the core application, a site wide configuration is stored in `<S_INSTALL_PATH>/var/slconfig`, and a user custom configuration is stored in `<HOME>/modellib`. If a global site configuration file exists for a given application and version it will override the factory default. Similarly, a user configuration overrides the site default.

Full details of the `slconfig` command and it use are described in Figure 1.

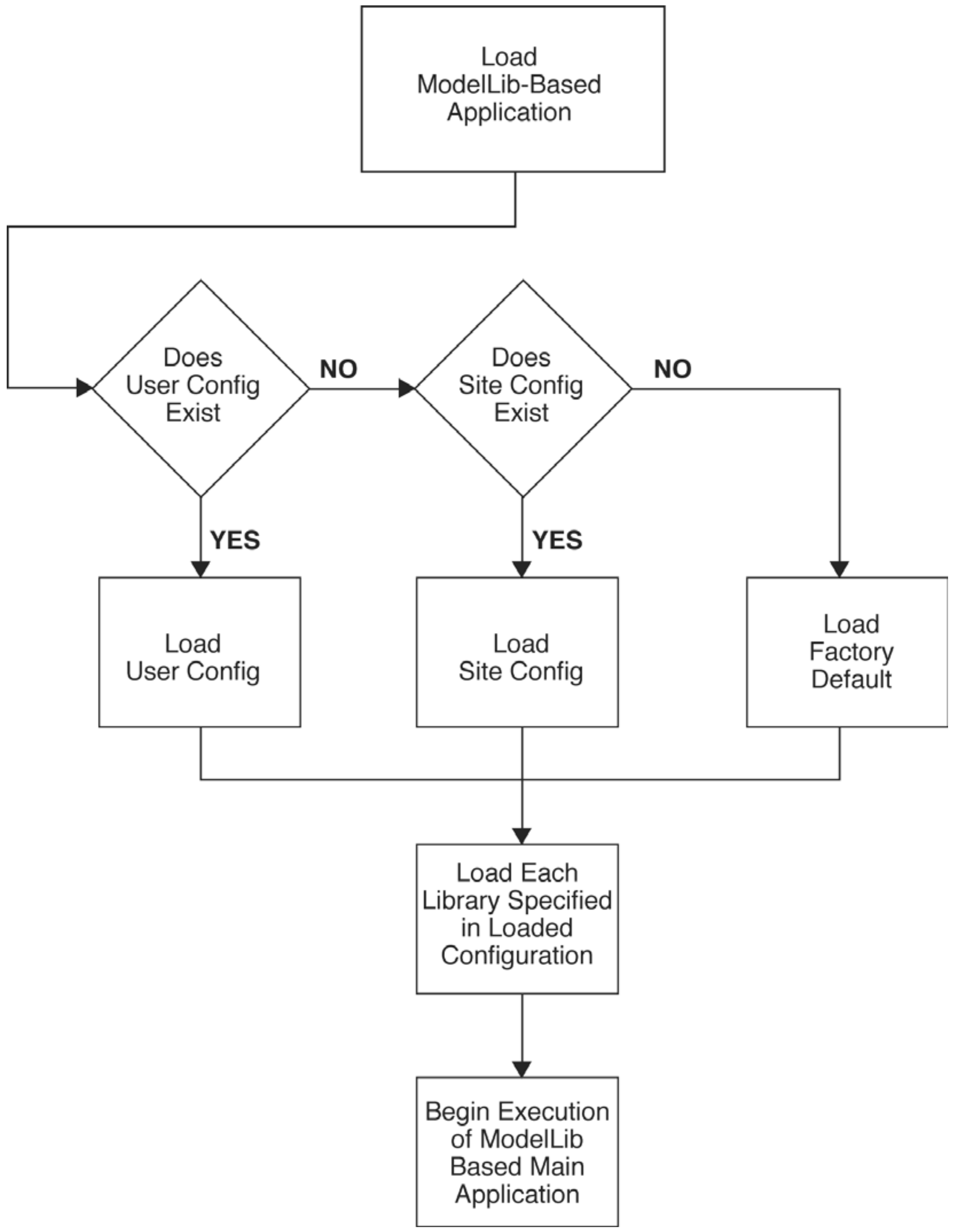

Load ModelLib-Based Application
Does User Config Exist
NO
Does Site Config Exist
NO
YES
YES
Load User Config
Load Site Config
Load Factory Default
Load Each Library Specified in Loaded Configuration
Begin Execution of ModelLib Based Main Application

Figure 1: Logic to Determine Which Configuration is Loaded

* The MODELLIB loader will pre-process the loaded configuration file to do environment substitutions. If `<my_env_variable>` exists in a file, it will be replaced with the contents of the environment variable `my_env_variable`. If the environment variable my_env_variable does not exist, `<my_env_variable>` is removed. Both path names and environment variable are case sensitive.

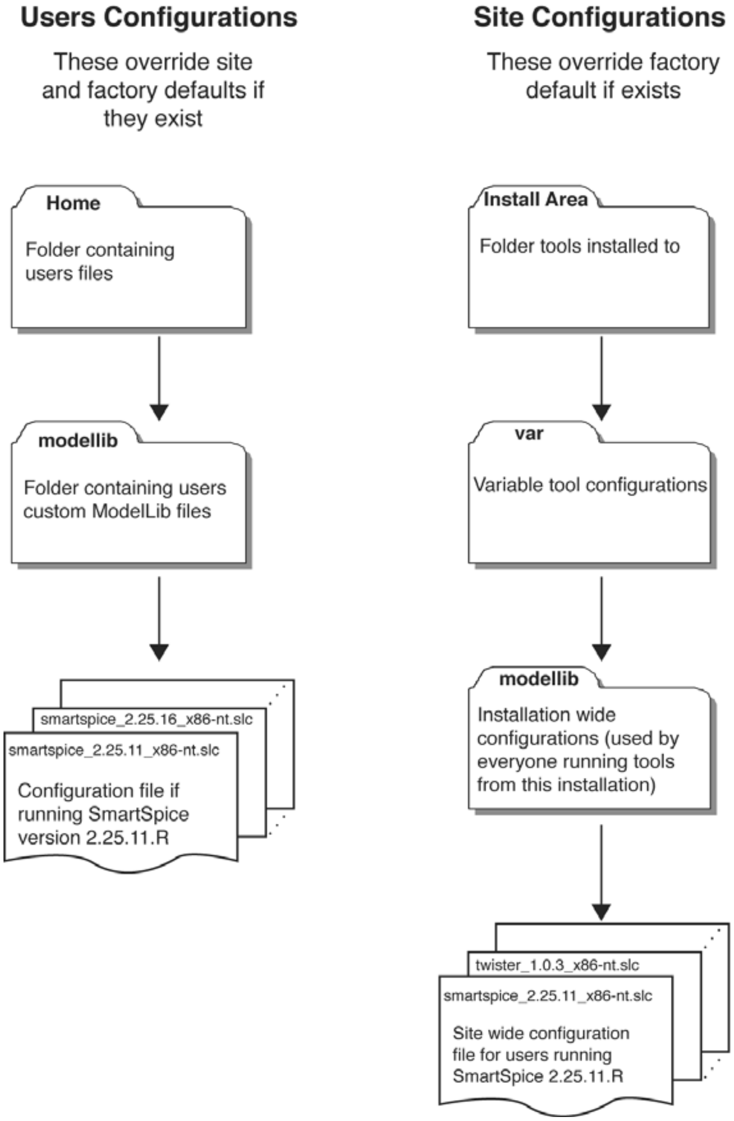

Users Configurations
These override site and factory defaults if they exist
Site Configurations
These override factory default if exists
Factory Defaults
Default Configuration
Home
Folder containing users files
modellib
Folder containing users custom ModelLib files
smartspice_2.25.16_x86-nt.slc
smartspice_2.25.11_x86-nt.slc
Configuration file if running SmartSpice version 2.25.11.R
Install Area
Folder tools installed to
var
Variable tool configurations
modellib
Installation wide configurations (used by everyone running tools from this installation)
twister_1.0.3_x86-nt.slc
smartspice_2.25.11_x86-nt.slc
Site wide configuration file for users running SmartSpice 2.25.11.R
SmartSpice Version 2.25.12.R Executable Binary
Factory Defaults ModelLib Configuration

Figure 2: ModelLib Configuration Files

ModelLib Installation Procedure

Binary models for MODELLIB are distributed in one of two forms:

1. A complete collection of models as part of a MODELLIB based application. This distribution contains the default set of all models needed by the given application. This distribution is included in the application download and is installed automatically when the application is installed.
2. Point updates of a SPICE model. This is an individual Model update and is distributed as a Standard Software Update (`.ssu`). These point updates are installed from `sman`, the Management Console program.

Installation of Binary Model Updates

The installation of a new SPICE model update is a two step process:

Step 1: Model Download and Installation

1. Download the new model from the Simucad Resource Center.
2. Type `sman` on Unix or Linux, or double-click the **Management Console** icon on Windows.
3. Select **Add** and **Manage Updates** from the Tools menu.
4. Click **Browse** and select the downloaded file.
5. Click **Install**.
6. The output will show the path and filename of the model installed.

Step 2: Configure a Tool to use a New Model (GUI Method)

1. Type `sman` on Unix or Linux, or double-click the **Management Console** icon on Windows.
2. Select MODELLIB from the Tools menu.
3. Select the **Product**, **Version** and **Platform** to configure.
4. Select the **User** or **Site** tab, and confirm the **copy from default** if promoted.
5. Select the model versions to be used from the table.
6. Click the **Save** button.

or

Step 2: Configure a Tool to use a New Model (CLI Method)

1. Type the `slconfig` command with the following parameters:

```
slconfig {ToolName} {ToolVersion} user replace {ModelName} '<S_INSTALL_ROOT>
/{ModelPath}'
```

Where:

- `{ToolName}` is the MODELLIB based tool you are configuring (e.g., `smartspice`).
- `{ToolVersion}` is the version of the tool that will use the new model (e.g., 2.25.12.R)
- `{ModelName}` is the name of the model being updated (e.g., `libBSIM3v3`).
- `{ModelPath}` is the path from the install directory to the model file (e.g., `lib/modellib/1.0.4.A/x86_64-linux/libBSIM3v3_1_0_4_R.so`).

Example

To install a newly downloaded version of `libBSIM3v3` (version 1_0_8_R) into your installed copy of SMARTSPICE: (version 2.25.12.R).

If the output of Step 1 gives the path as:

```
lib/modellib/1.0.4.A/x86_64-linux/libBSIM3v3_1_0_4_R
```

type (all on one line):

```
slconfig smartspice 2.25.12.R user replace libBSIM3v3 '<S_INSTALL_ROOT>/lib/
modellib/1.0.4.A/x86_64-linux/libBSIM3v3_1_0_4_R.so'
```

Note: The quotes around the last argument allow the string `<S_INSTALL_ROOT>` to be used. This string is automatically set to the location where your tools are installed.

GUI Reference (SMAN)

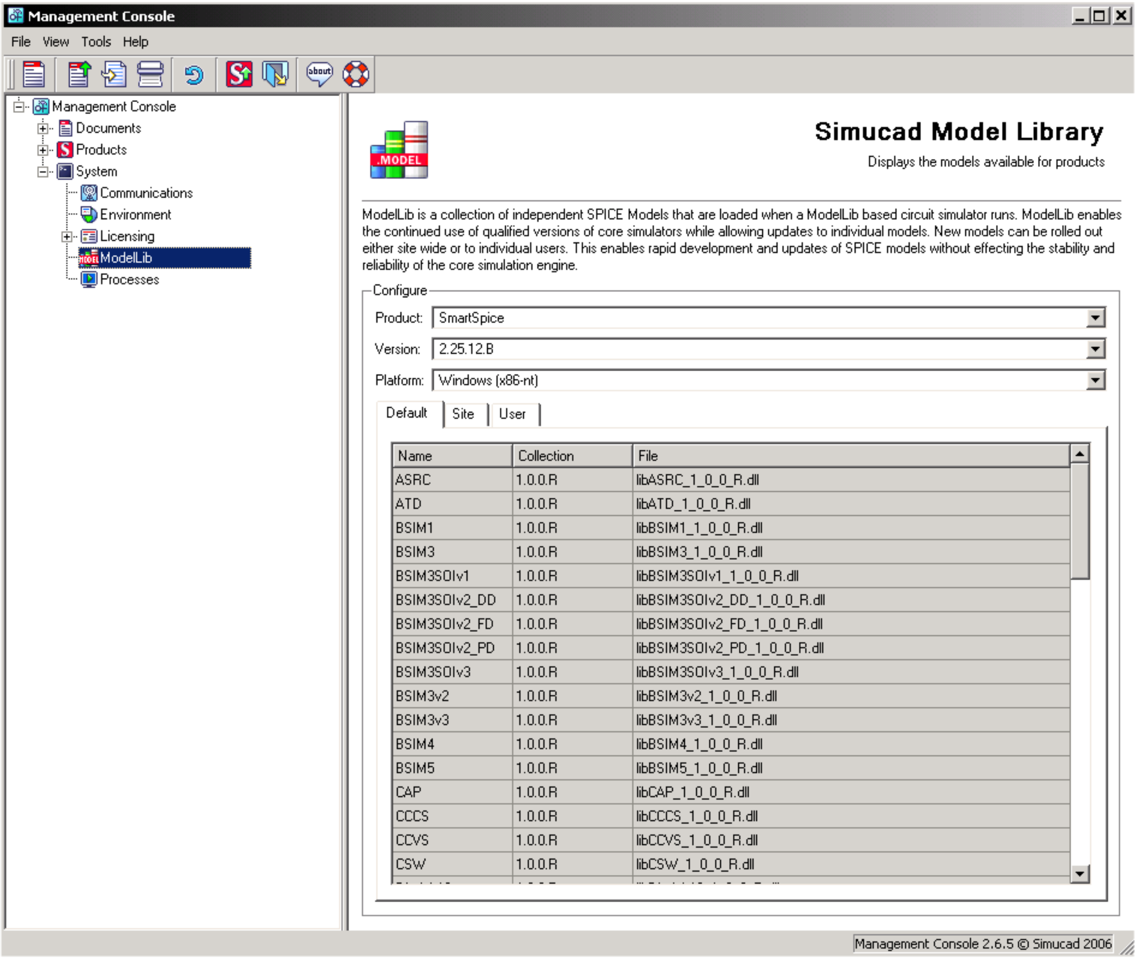
To access the GUI configuration:

1. Type `sman` on Unix or Linux, or double-click the **Management Console** icon on Windows.
2. Select MODELLIB from the Tools menu.

Simucad Model Library Screen

The Simucad Model Library (MODELLIB) Screen (Figure 3) allows you to configure each of the model library enabled products. The configuration is divided into three sections the **Default**, **Site** and **User**.

The **Default** configuration contains a list of the default models that are loaded by the model library enabled products. However, the **Default** configuration can not be edited or modified in any way. If you want to override the default models, or add custom user models, you can change or override the default configuration for the entire site, or just for an individual user.



Simucad Model Library

Displays the models available for products

ModelLib is a collection of independent SPICE Models that are loaded when a ModelLib based circuit simulator runs. ModelLib enables the continued use of qualified versions of core simulators while allowing updates to individual models. New models can be rolled out either site wide or to individual users. This enables rapid development and updates of SPICE models without effecting the stability and reliability of the core simulation engine.

Configure

Product: SmartSpice

Version: 2.25.12.B

Platform: Windows (x86-nt)

Default | Site | User

Name	Collection	File
ASRC	1.0.0.R	libASRC_1_0_0_R.dll
ATD	1.0.0.R	libATD_1_0_0_R.dll
BSIM1	1.0.0.R	libBSIM1_1_0_0_R.dll
BSIM3	1.0.0.R	libBSIM3_1_0_0_R.dll
BSIM3SOIv1	1.0.0.R	libBSIM3SOIv1_1_0_0_R.dll
BSIM3SOIv2_DD	1.0.0.R	libBSIM3SOIv2_DD_1_0_0_R.dll
BSIM3SOIv2_FD	1.0.0.R	libBSIM3SOIv2_FD_1_0_0_R.dll
BSIM3SOIv2_PD	1.0.0.R	libBSIM3SOIv2_PD_1_0_0_R.dll
BSIM3SOIv3	1.0.0.R	libBSIM3SOIv3_1_0_0_R.dll
BSIM3v2	1.0.0.R	libBSIM3v2_1_0_0_R.dll
BSIM3v3	1.0.0.R	libBSIM3v3_1_0_0_R.dll
BSIM4	1.0.0.R	libBSIM4_1_0_0_R.dll
BSIM5	1.0.0.R	libBSIM5_1_0_0_R.dll
CAP	1.0.0.R	libCAP_1_0_0_R.dll
CCCS	1.0.0.R	libCCCS_1_0_0_R.dll
CCVS	1.0.0.R	libCCVS_1_0_0_R.dll
CSW	1.0.0.R	libCSW_1_0_0_R.dll

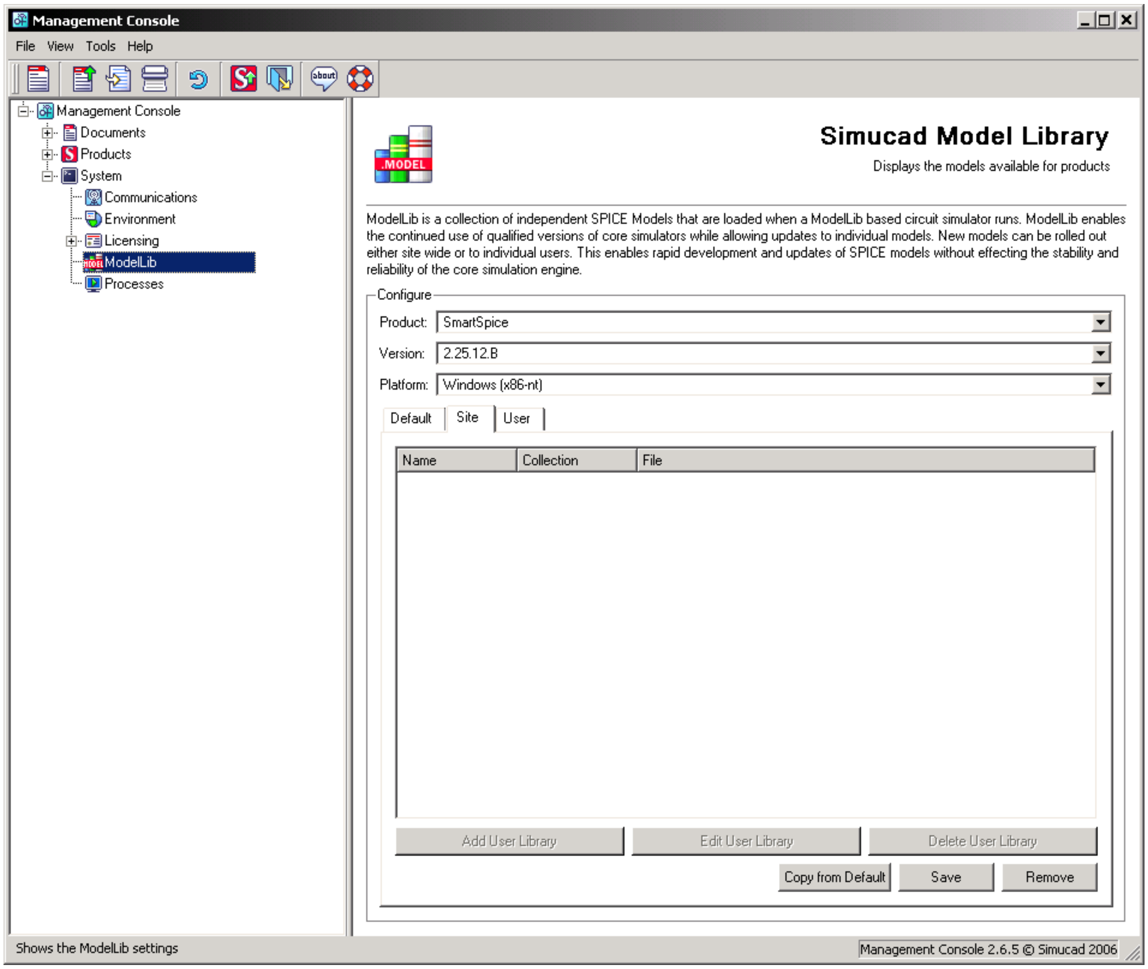
Management Console 2.6.5 © Simucad 2006

Figure 3: ModelLib Screen

Site Configuration

The **Site** configuration allows you to specify the models that will be applied to all users that run a MODELLIB enabled product from this installation directory. The configuration file is placed in `<S_INSTALL_ROOT>/var/slconfig` where `<S_INSTALL_ROOT>` is the root directory where the product have been installed (i.e., `c:\sedatools`).

To manage the site configuration, you must first select the **Site** tab in the screen, and then select the product that is to be configured. The list of model library enabled products is controlled by the **Product** combo-box. Once you have selected the product that is to be configured, the list of available versions to configure will be populated in the **Version** combo-box. Finally the list of available platforms to configure is populated in the **Platform** combo-box. If you have installed the products to a platform specific folder, then only the platform that is available will be shown.



ModelLib is a collection of independent SPICE Models that are loaded when a ModelLib based circuit simulator runs. ModelLib enables the continued use of qualified versions of core simulators while allowing updates to individual models. New models can be rolled out either site wide or to individual users. This enables rapid development and updates of SPICE models without effecting the stability and reliability of the core simulation engine.

Product: SmartSpice

Version: 2.25.12.B

Platform: Windows (x86-nt)

Name	Collection	File

Shows the ModelLib settings

Management Console 2.6.5 © Simucad 2006

Figure 4: Site Configuration

If no **Site** configuration exists you will be prompted to copy one from the default configuration. The starting configuration can also be regenerated at a later date by pressing the **Copy from Default** button.

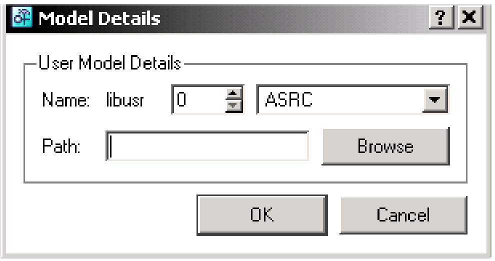
Once the **Site** tab contains model libraries they can be modified, or you can choose to add custom user libraries.

To modify an existing library, change the collection that it is pointing to (each collection refers to a different version of the MODELLIB libraries, which contain the default libraries released by Simucad) or change the model file that it is using. You may only pick model files that are compatible with the model type (i.e., you cannot choose a BSIM model for a TFT model).

Warning: Changing the collection is not recommended. You should always choose models from the same collection as the default where possible not doing so may cause instability.

Alternatively, you may add, edit or delete a custom user library by pressing the **Add User Library**, **Edit User Library** or **Delete User Library** buttons respectively.

When adding or editing a custom user library the **Model Details** dialog (Figure 5) will show.



Model Details

User Model Details

Name: libusr 0 ASRC

Path:

Figure 5: Custom User Library Details

You may only delete a custom user library, and if you press the **Delete User Library** button while a default library is selected, the action will be ignored.

Once you are happy with the site configuration, the changes can be committed by pressing the **Save** button. This will create a site configuration file in the `<S_INSTALL_ROOT>/var/slconfig` directory, where `<S_INSTALL_ROOT>` is the root directory where the product have been installed (i.e., `c:\sedatools`).

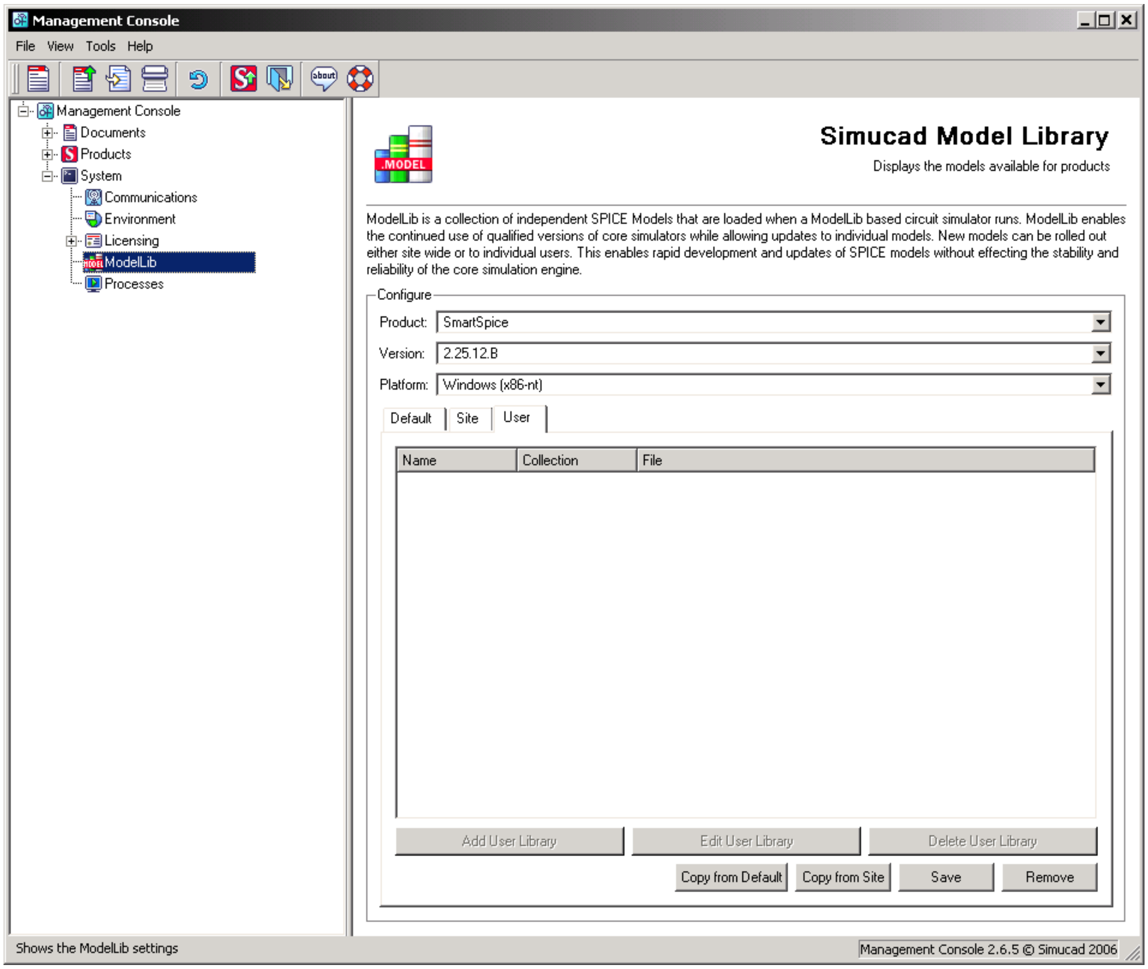
If you want to reset the site configuration, press the **Remove** button, which will remove the site configuration file and clear the entries in SMAN.

User Configuration

The **User** configuration allows you to specify the models that will that a specific user will use when running a specific version of a MODELLIB enabled product. The **User** configuration will override both the default and site configurations. The configuration file is placed in `<HOME>/slconfig` where `<HOME>` is the users home directory. On Windows, the home directory is determined by the following logic in order of precedence:

1. Use the environment variable `HOME`.
2. Use the environment variable `USERPROFILE`.
3. Use the path formed by concatenating the `HOMEDRIVE` and `HOMEPATH` environment variables.
4. Use the environment variable `SYSTEMDRIVE`.
5. Default to `C:\`

To configure the models for a user you must first choose the product that is to be configured. The list of model library enabled products is controlled by the **Product** combo-box. Once you have selected the product that is to be configured, the list of available versions to configure will be populated in the **Version** combo-box. Finally the list of available platforms to configure is populated in the **Platform** combo-box. If you have installed the products to a platform specific folder, then only the platform that is available will be shown.



Simucad Model Library

Displays the models available for products

ModelLib is a collection of independent SPICE Models that are loaded when a ModelLib based circuit simulator runs. ModelLib enables the continued use of qualified versions of core simulators while allowing updates to individual models. New models can be rolled out either site wide or to individual users. This enables rapid development and updates of SPICE models without effecting the stability and reliability of the core simulation engine.

Configure

Product: SmartSpice

Version: 2.25.12.B

Platform: Windows (x86-nt)

Shows the ModelLib settings

Management Console 2.6.5 © Simucad 2006

Figure 6: User Configuration

If no **User** configuration exists, you will be prompted to copy one from the default configuration or site configuration. The starting configuration can also be regenerated at a later date by pressing the **Copy from Default** or **Copy from Site** buttons.

Once the **User** tab contains model libraries, they can either be modified, or you can choose to add custom user libraries.

To modify an existing library, change the collection that it is pointing to (each collection refers to a different version of the MODELLIB libraries which contain the default libraries released by Simucad), or change the model file that it is using. You may only pick model files that are compatible with the model type (i.e., you cannot choose a BSIM model for a TFT model).

Warning: Changing the collection is not recommended. You should always choose models from the same collection as the default where possible not doing so may cause instability.

Alternatively, you may add, edit or delete a custom user library by pressing the **Add User Library**, **Edit User Library** or **Delete User Library** respectively.

When adding or editing a custom user library the **Model Details** dialog (Figure 5) will appear.

You may only delete a custom user library, and if you press the Delete User Library button as a default library is selected, the action will be ignored.

Once you are happy with the user configuration, the changes can be committed by pressing the **Save** button. This will create a user configuration file in the `<HOME>/slconfig` directory, where `<HOME>` is the users home directory (see the previous sections for Windows configuration).

If you want to reset the user configuration then press the **Remove** button, which will remove the user configuration file and clear the entries in SMAN.

Command Reference (slconfig)

Name

`slconfig`: Configure applications to use updated MODELLIB Models

Synopsis

```
slconfig {ToolName} {ToolVersion} user|site|default list
slconfig {ToolName} {ToolVersion} user|site add {ModelName} {ModelPath}
slconfig {ToolName} {ToolVersion} user|site replace {ModelName} {ModelPath}
slconfig {ToolName} {ToolVersion} user|site remove {ModelName}
slconfig {ToolName} {ToolVersion} user|site copy default|site
slconfig {ToolName} {ToolVersion} user|site delete
```

Description

The `slconfig` utility creates, updates and removes configurations for MODELLIB based applications. The configurations are used by the application to determine what dynamic model libraries are loaded at runtime. `slconfig` can be used to manipulate either site wide configuration, or a users own configuration for a given tool and version.

Every tool ships with a default configuration that will be loaded and used if no other custom configurations have been made. If a suitable site wide configuration exists, the tool will load that configuration in preference to the default configuration. Likewise, if a suitable user specific configuration exists, that configuration will be used in preference to either the site wide or default configuration. A configuration is suitable if the configuration file matches the name of the tool, the version of the tool and the platform.

The MODELLIB loader will pre-process the loaded configuration file to do environment substitutions. When specifying a model path, an environment variable can be used as part of the path by enclosing it in < and >. For example `<HOME>/mymodels/mymodel.so` will place the contents of the HOME environment variable into the path. Similarly, If `<my_env_variable>` exists in a file, it will be replaced with the contents of the environment variable `my_env_variable`. If the environment variable my_env_variable does not exist `<my_env_variable>` is removed.

Note: When using environment variables as part of the path name specification it is necessary to enclose the entire path in single quotes `'` and `'` to prevent the < and > from being interpreted as file redirection by the command `shell`.

All Simucad tools automatically configure the environment variable `S_INSTALL_ROOT` as the location of the installed software. This variable must never be set directly by a user, but can still be used in `slconfig` to make the configuration location independent.

Configuration files managed by `slconfig` must never be edited by hand.

Options

- {ToolName} Is the name of the tool that is being configured (e.g., SMARTSPICE).

This tool must be one of the Tool names in the following table.

ToolName	Description
smartspice	SMARTSPICE circuit simulator, use also for SMARTSPICERF.
twister	Fast Circuit simulator.

- {ToolVersion} Is the version of the tool being configured in standard three number plus letter format (Num.Num.Num.Letter). For example, 2.25.12.R.

The versions of a tool installed can be determined simply by using the -WVs option to the tool. For example:

```
smartspice -WVs lists the versions of smartspice installed.
```

- user specifies that slconfig will operate on the user's own configuration.
- site specifies that slconfig will operate on the global site configuration.
- default specifies that slconfig will operate on the factory default configuration. This configuration can only be listed and not edited.
- {modelname} The internal name by which the MODELLIB based application will refer to the model. See Standard Model Names for a list of the standard model names.
- {modelpath} The file path to the model. The model path will be preprocessed to do environment substitutions. For example, if <my_env_variable> exists in the path, it will be replaced with the contents of the environment variable my_env_variable. If the environment variable my_env_variable does not exist, <my_env_variable> is removed from the path.

Operations

add	Adds a new model to the configuration file. This command takes the two options ModelName and the Path to the library component to load.
replace	Replaces an existing Model with a new Model. This command takes the two options ModelName and the Path to the library component to load.
remove	Removes a model from the configuration file. This command takes one option ModelName to remove.
list	Lists the contents of the configuration.
copy	Copies either the factory default or site default configuration file. slconfig {ToolName} {ToolVersion} site sitedefault does nothing.
delete	Deletes the configuration file completely.

Examples

Replace the libBSIM3v3 used by SMARTSPICE 2.25.12.R with a new model installed in the tools folder as:

```
lib/modellib/1.0.4.A/x86_64-linux/libBSIM3v3_1_0_8_R.so
slconfig smartspice 2.25.12.R user replace libBSIM3v3 '<S_INSTALL_ROOT>/lib/
modellib/1.0.4.A/x86_64-linux/libBSIM3v3_1_0_8_R.so'
```

List the contents of TWISTER 1.4.4.R factory defaults:

```
slconfig twister 1.4.4.R default list
```

Remove the site configuration for SmartSpice 2.25.12.R:

```
slconfig smartspice 2.25.12.R site delete
```

Add a custom model to SmartSpice 2.25.12.R; the model is `libFooBar` located at:

```
$HOME/mymodels/libMyModel.so.
slconfig smartspice 2.25.12.R user add libMyModel '<HOME>/mymodels/libMyModel.so'
```

Remove an experimental Model `libMyModel` from the users local configuration for SmartSpice 2.25.12.R:

```
slconfig smartspice 2.25.12.R user remove libMyModel
```

Standard Model Names

The Following table is a list of the standard model names distributed with ModelLib.

```
libASRC
libBSIM1
libBSIM3SOIv2_DD
libBSIM3SOIv2_FD
libBSIM3SOIv2_PD
libBSIM3SOIv3
libBSIM3
libBSIM3v3
libBSIM4
libBSIM5
libBSMG
libBSMG3T
libCAP
libCCCS
libCCVS
libCSW
libDiodeL13
libDiodeL2
libDiodeL500
libEKV
libFRMC
libHBT
libHICUM
libHiSIM
libHiSIMHU
libHiSIMHULDMOS
libHVMOS
libIND
libISRC
libJFETMESFET
libJuncap
libLETISOI
libLTRA
libMEXTRAM
libMODELLA
libMOSlevel123
libMOS11
libMOS20
libMOS31
libMUT
libNPORTS
libPORT
libPSP
libPSPBIN
libRES
libRPIaSi
libRPIpolySi
libSGP
libSW
libTRA
libTXL
libURC
libVBIC
libVCCS
libVCSEL
libVCVS
libVSRC
libWTRA
```

Files

```
<S_INSTALL_DIR>/var/slconfig contains global configurations
<HOME>/slconfig contains users own configurations
```

See also:

```
sman
```

Note: ModelLib configuration files must never be hand edited; `slconfig` must be used as the format, may contain binary data, and can be changed without notice.

ModelLib Models and Corresponding Shared Libraries

Type Group	Technology	Internal Name	Info	Level	Lib Name
npn pnp lpnp	BJT	BJT	Bipolar Junction Transistor	1, 2	libSGP
		VBIC	VBIC Bipolar Junction Transistor	4	libVBIC
		HICUM	HICUM Bipolar Junction Transistor	8	libHICUM
		PBJT	Mextram BJT (Philips)	6,503,504	libMEXTRAM
		MODELLA	Philips TPL500 Bipolar Transistor	500	libMODELLA
		HBT	Hetero-Junction Bipolar Transistor	20	libHBT
nmos pmos ntft ptft	SOI	BSIM3SOI2DD	Berkeley SOI MOSFET model version 2 (level 27)	27	libBSIM3SOIv2_DD
		BSIM3SOI2FD	Berkeley SOI MOSFET model version 2 (level 26)	26	libBSIM3SOIv2_FD
		BSIM3SOI2PD	Berkeley SOI MOSFET model version 2 (level 29)	29	libBSIM3SOIv2_PD
		BSIM3SOI3	Berkeley SOI MOSFET model version 3 (level 33)	33, 57	libBSIM3SOIv3
		LETISOI	CEA/LETI SOI MOSFET model 32	32	libLETISOI
	TFT	MOS15	RPI AmorphousSi TFT Model	35	libRPIaSi
		MOS16	RPI Poly-Si TFT Model	36	libRPIpolySi

Type Group	Technology	Internal Name	Info	Level	Lib Name
	MOSFET	MOS123	MOS field-effect transistor	1, 2, 3	libMOSlevel123
		BSIM1	Berkeley Short Channel IGFET Model	4, 13	libBSIM1
		BSIM3	Berkeley Short Channel IGFET Model Version-3 (level 81)	81	libBSIM3
		BSIM3v3	Berkeley Short Channel IGFET Model Version-3 (level 8, 11, 49, 53)	8, 11, 49, 53	libBSIM3v3
		BSIM4S	Berkeley Short Channel IGFET Model-4 (level 14, 54)	14, 54	libBSIM4
		BSIM5	Berkeley Short Channel IGFET Model-5	55	libBSIM5
		BSMG	Berkeley BSIM Multi-Gate MOSFET model	51	libBSMG
		BSMG3T	Berkeley BSIM Multi-Gate MOSFET model 3-terminal	52	libBSMG3T
		MOS11	Philips MOS11 model	63	libMOS11
		MOS31	MOS31 MOSFET Model	30, 31, 40	libMOS31
		MOS20	Philips MOS20 LDMOS model	20	libMOS20
		EKV	EKV MOSFET Model	44	libEKV
		BSIM3H	High-Voltage MOSFET Model (level 88)	88	libHVMOS

Type Group	Technology	Internal Name	Info	Level	Lib Name
		HISIM	Hiroshima University STARC IGFET Model (level 70)	70	libHiSIM
		HISIMHU	Hiroshima University STARC IGFET Model (level 170)	170	libHiSIMHU
		HISIMHULD-MOS	HiSIM-LDMOS Model	172	libHiSIMHULD-MOS
		PSP	Philips PSP MOSFET model	56	libPSP
		PSPBIN	Philips PSP MOSFET model with binning (level 58)	58	libPSPBIN
njf pjf nmf pmf	JFET/ MESFET	JFET	Junction/Schottky contact field-effect transistor	1, 2, 3, 4, 5, 6	libJFETMESFET
d	Diode	DIO	Junction Diode	1, 3	libDiodeL13
		DIO2	Fowler-Nordheim Diode	2	libDiodeL2
		DIO500	Diode Level 500	500	libDiodeL500
		JCAP	Junction Capacitor	9, 10	libJuncap
		LAS1	VCSEL model	4	libVCSEL
Passive devices	Capacitances	FRMC	Ramtron Ferroelectric Capacitance Model	6	libFRMC
		CAP	Capacitance	1,3	libCAP
	Resistor	RES	Resistor		libRES
	Inductors	IND	Inductor		libIND
		MUT	Mutual inductor		libMUT
Multiterminal	Port	NPORT	Multiterminal port		libNPORTS

Type Group	Technology	Internal Name	Info	Level	Lib Name
Independent sources	Voltage	VSRC	Voltage source		libVSRC
	Current	ISRC	Current source		libISRC
	Port	PORT	Resistive source port		libPORT
Controlled sources	Current	CCCS	Current controlled currents		libCCCS
	Voltage	CCVS	Current controlled voltages		libCCVS
	Current	VCCS	Voltage controlled currents		libVCCS
	Voltage	VCVS	Voltage controlled voltages		libVCVS
	Arbitrary	ASRC	Arbitrary source		libASRC
Switches	Current	CSW	Current controlled switch		libCSW
	Voltage	SW	Voltage controlled switch		libSW
Transmission lines		TRA	Lossless transmission line		libTRA
		LTRA	Lossless transmission line		libLTRA
		WTRA	W transmission line		libWTRA
		URC	U transmission line		libURC
		TXL	Uncoupled lossy trans. line		libTXL

[This page intentionally left blank.]