AMECustom

Version 4.2 - September 2004



Copyright © IMAGINE S.A. 1995-2004

AMESim® is the registered trademark of IMAGINE S.A.

AMESet® is the registered trademark of IMAGINE S.A.

ADAMS® is a registered United States trademark of Mechanical Dynamics, Incorporated.

ADAMS/SolverTM and ADAMS/ViewTM are trademarks of Mechanical Dynamics, Incorporated.

MATLAB and SIMULINK are registered trademarks of the Math Works, Inc.

Netscape and **Netscape Navigator** are registered trademarks of Netscape Communications Corporation in the United States and other countries. Netscape's logos and Netscape product and service names are also trademarks of Netscape Communications Corporation, which may be registered in other countries.

PostScript is a trademark of Adobe Systems Inc.

UNIX is a registered trademark in the United States and other countries exclusively licensed by X / Open Company Ltd.

Windows, Windows NT, Windows 2000, Windows XP and Visual C++ are registered trademarks of the Microsoft Corporation.

The **GNU Compiler Collection (GCC)** is a product of the Free Software Foundation. See the GNU General Public License terms and conditions for copying, distribution and modification in the license file.

X windows is a trademark of the Massachusetts Institute of Technology.

All other product names are trademarks or registered trademarks of their respective companies.

Table of contents

Chapter 1: In	troducing AMECustom1
1.1	Presentation of this manual
1.2	What is AMECustom?1
1.3	Customized submodels
1.4	Customized supercomponents
1.5	Important rules for customized objects
	Customized submodels5Customized supercomponents5Encryption5Protection of encrypted objects5Distribution of systems containing customized objects5
Chapter 2: Cu	ustomizing submodels
2.1	Introduction
2.2	Example 1: customized submodel of the cosine function7
2.3	Files associated with a customized submodel15
2.4	Example 2: creating two customized submodels from a cam generic
	submodel
Chapter 3: Co	ustomizing supercomponents
3.1	Example 1: A customized cam supercomponent
3.1.	1 Constructing the flat system
3.1.2	2 Constructing the generic supercomponent
3.1.	3 Customizing the supercomponent
3.2	Example 2: A 2-level customized supercomponent with global
	parameters
3.2.	1 Introduction
3.2.2	2 Constructing the flat system
3.2.3	Constructing the customized supercomponent
3.2.4	4 Activity variables and customized supercomponents
3.3	Conclusion
Chapter 4: Ro	eference Guide for AMECustom
4.1	Introduction

4.2 T	he AMECustom main window	47
4.3 T	he AMECustom menu bar	49
4.3.1	File menu	49
	Open	49
	Save	50
	Save as.	50
	Last opened files list	
432	Edit menu	51
1.5.2	Cut	52
	Copy	
	Paste	52
	Delete	52
	Add new	52
	Trash can	53
	Move up	53
	Hide all	55
	Show all	
	Update categories	53
	External variables	54
	Available customized	54
	Available supercomponents	54
122	Available user submodels	
4.3.3		
	Path list	
	AMECustom Preferences	57
4.3.4	Previews menu.	58
	Parameter mode	58
	Run mode	58
4.3.5	Icons menu.	58
4.3.6	Tools menu	62
	AMELexicon	63
	Check submodels	66
	Check submodels of a customized submodel	66
	Check submodels of a customized supercomponent	67
	Expression Editor	69
	Start AMESim/Start AMESet	
437	Documentation menu	/1
л. <i>э.</i> т Д 2 0	Windows manu	···· 12
4.3.8		/3
	Cascade	73
	List of currently opened files	13

close all4.3.9 Help menu	
Online	73
About	74
4.4 The AMECustom Toolbar	74
4.5 The Category buttons	75
4.6 The customized object general features	77
4.6.1 General features of a submodel	77
Name	77
Icon	77
Brief description	78
Full description	
4.6.2 General features of a supercomponent	/9
Name, Icon, Brief description, Full description	80
List of constituents.	80
Main functions available from the right-click menus	
4.7 The global parameter lists	85
Real	86
Integer	87
Text	88
Using global parameters	
4.8 The External variables list	91
External variable characteristics	
4.9 The Internal items lists	92
Internal variables	03
Real parameters	
Integer parameters	
Text parameters	94
Right-click menus	95

Table of contents

Chapter 1: Introducing AMECustom

1.1 Presentation of this manual

The **AMECustom** user manual will help you to:

- understand the aim of AMECustom and what you can do with this complementary application of AMESim,
- learn how to customize submodels and supercomponents.

This user manual is split up into four chapters:

Chapter 1 introduces **AMECustom** and customized objects.

Chapter 2 gives two examples of customized submodels.

Chapter 3 gives two examples of customized supercomponents.

Chapter 4 is a reference chapter which can be used from the main index.

We recommend that you go through these examples before you attempt your own customizations.

1.2 What is AMECustom?

Before answering this question it is worth reviewing what you can do with each product of the **AMESim** suite.

AMESim

You build models by connecting together components either directly or using lines. In *Submodel* mode you attach mathematical descriptions to each component and line. These are either submodels or supercomponents. You can then change parameters, perform runs and analyze results.

You can also develop your own supercomponents.

AMERun

You cannot build models and supercomponents. But you can use models built from **AMESim** for changing parameters, performing runs and analyzing results.

AMESet

You create your own submodels for using them in **AMESim** models.

The submodels you create in **AMESet** and the supercomponents you create in **AMESim** are described as **generic**. When submodels are of a very general nature, the number of parameters to set can be very large. This is even more true for supercomponents. To overcome this problem the **customized** submodels and supercomponents were introduced.

AMECustom

In **AMECustom** you load generic objects (submodels or supercomponents) and then:

- fix and hide certain parameters,
- rename other parameters and set new default values,
- link together two or more parameters with a global parameter having a scope limited to the customized object,
- hide certain variables,
- retitle other variables and set new default values.

The result is a customized version of the generic object.

1.3 Customized submodels

An example of very simple level of customization of submodels is in the *Thermal Hydraulic* library. The fluid properties submodels *TFFD1* and *TFFD2* are generic submodels and associated with them are 10 customized versions of *TFFD1* and 10 customized versions of *TFFD2*.

Figure 1.1 shows the generic TFFD1 with its 10 customized versions.

Submodel List thf_fluid_data Currently compor submodel TFFD1 Submodel list	• TFFD1
Name TF-PURE_WATER TF_PURE_GLYCOL TF_DIESEL_SIMPLE TF_COOLANT_EG20W8 TF_COOLANT_EG40W6 TF_COOLANT_EG80W2 TF_COOLANT_EG60W4 TF_COOLANT_SIMPLE TF_COLANT_SIMPLE TF_COL_15W40 TF_OIL_15W30	Description thermal-hydraulic simple properties (polynomials)
Copy common parameters whe	n submodel changes Explore Bemove <u>QK</u> Cancel

Figure 1.1: TFFD1 has 10 customized versions



Figure 1.2: Generic submodel TFFD1 in Parameter mode

😽 Change Parameters 🔋		
Submodel TFFD1-1 thermal-hydraulic (polynomials)	simple properties	<u>x</u> ternal variables
Available customized submodels: Copy common parameters when sul Parameters	TFFD1 bmodel changes	T
Title index of thermal hydraulic fluid initial temperature name of the fluid filename for fluid characteristic data	Value \$AME/libthh/data/coc	Unit 1 20 degC coolant ilant.data
<u>S</u> ave Load	Default value	Ma <u>x</u> , value Mi <u>n</u> , value
<u>H</u> elp	<u>C</u> ancel	Options >>

A customized version is shown in Figure 1.3.

Figure 1.3:Customized ve	ersion in I	Parameter	mode
--------------------------	-------------	-----------	------

🙀 Change Parameters	? ×
Submodel TF_PURE_WATER-1 thermal-hydraulic simple p (polynomials)	I E <u>x</u> ternal variables properties
Available customized submodels: Copy common parameters when submodel of Parameters Title Value Un index of thermal hydraulic fluid 1 initial temperature 20 deg	TF_PURE_WATER TF_PURE_WATER TF_PURE_GLYCOL TF_DIESEL_SIMPLE TF_COOLANT_EG40W60 TF_COOLANT_EG80W20 TF_COOLANT_EG80W40 TF_COOLANT_EG60W40 TF_COOLANT_SIMPLE
<u>Save</u>	TF_OIL_15W40
Help	<u>OK</u> <u>C</u> ancel Options >>

Note the pulldown menu allowing switching between generic submodel and the 10 customized versions in *Parameter* mode. This does not involve any recompilation and it can be done in **AMERUN** as well as **AMESIM**. Note also the parameter list is shorter and simpler for the customized submodels.

This level of customization is very light but it is powerful and very valuable especially in **AMERun**. All that has happened is that in each customized versions, 2 parameters of the generic submodel are hidden. One of them is fixed to the name of a data file which supplies the information on the fluid concerned.

1.4 Customized supercomponents

When you customize a supercomponent, you make it appear in **AMESim** and **AMERun** as if it was a submodel. You can explore the contents of the customized supercomponent only in **AMECustom**, but if in **AMECustom** you saved it encrypted, you must supply the correct password to reopen it.

Customization can be multi-level: customized supercomponents can contain other customized objects.

1.5 Important rules for customized objects

Comments in this section are important but you may not fully understand their significance at a first reading. We suggest you reread it after completing the tutorial examples.

Customized submodels

For using a customized submodel in an **AMESim** model, the corresponding generic submodel must be available, and in particular its compiled code (*.o* or *.obj*) must exist.

Customized supercomponents

A customized supercomponent can be used in an **AMESim** model even if the corresponding generic supercomponent is deleted or unavailable.

Encryption

The **encryption** of a customized object in **AMECustom** does not affect anything in **AMESim**. However:

- If you look in the *.spe* and *.sub* files with an editor, you will find them rather difficult to understand!
- If you look in the .*cir* file created by **AMESim**, you will find that its relevant sections are also encrypted.
- Using **AMECustom**, when you try to open an encrypted customized object, you must enter the correct password before getting access to it.

Protection of encrypted objects

Working out the internal details of an encrypted customized submodel is possible given time. Concerning an encrypted customized supercomponent, without the corresponding generic supercomponent, it would be extremely difficult.

Distribution of systems containing customized objects

AMESIM is increasingly used as a model exchange software. Customization with encryption is an important part of this process.

Suppose you send a compiled **AMESim** system containing one or more customized objects to another user. The *AMEPack* facility in **AMESim** is designed to make this easy.

Chapter 1 Introducing AMECustom

The user can run the system in **AMERUN**. Using **AMESim** it is also possible to run the system, but if the executable has to be recreated, **the following files are needed**:

- The object files of the generic submodels corresponding to the customized submodels,
- The object files of the generic submodels used in the customized supercomponent constituents.

However the following files are NOT strictly needed:

- Any .spe files and .c (or .f) files of generic submodels.
- Any .spe files of customized submodels.
- Any .spe and .sub files of any generic or customized supercomponents.

If confidentiality is important, use the above information to minimize the files you send in the package.

Chapter 2: Customizing submodels

2.1 Introduction

This chapter shows you how to customize **AMESim** submodels with simple tutorial examples. However, before attempting to read this chapter and doing the exercises, you should have experience at using **AMESim** and performing simple simulations with standard **AMESim** submodels.

Name conventions

For generic submodels and supercomponents the following restrictions apply to the names:

- Names must be of 4 to 23 characters comprising uppercase letters and digits.
- The first character must be a letter.

For customized objects the rules are the same except that the characters '-' (minus) and '_' (underscore) are permitted. It is a good idea to use these freely so as to make customized objects distinctive.

Please note also that:

- In all the **AMESim** libraries, standard submodels and supercomponents have the first digit in their name (if any) in the range **0** to **4**, whether they are generic or customized.
- Hence, if you create your own customized objects with names that contain at least one digit and if the first digit is in the range 5 to 9, there will be no risk of name clashes with existing standard objects.

2.2 Example 1: customized submodel of the cosine function

Objective:

• To create the simplest possible customized submodel.

The function submodel FX00 belongs to the *Signal, Control and Observers* library. The icon it is associated with is shown. This submodel applies any user-supplied function of one variable x, to the input signal on port 2. The result is the output f(x) on port 1. The function *f* is defined by the user in *Parameter* mode.



A simple customization can be made to FX00 to produce COS50, which will calculate the cosine of the input. We could also produce a sine version, an absolute value version and indeed a whole collection of customized versions of FX00.

For this simple example you will load the generic submodel *FX00* into **AMECustom** and customize it saving the resulting customized submodel as *COS50*. It is a good idea to create a special directory or folder called *tutorial* (for instance) and save all the customized submodels in this directory.

Step 1: Start AMECustom

Using Unix:

Talk to your system administrator who will show you how to set up your working environment so that you get access to **AMECustom**. To start **AMECustom**, in a suitable window change to the directory where you wish to work (*tutorial* for instance) and type:

AMECustom

Using Windows:

Do one of the following:

- Select *AMECustom* from the menu **Program** ► **Imagine** ► **AMECustom** produced by the **Start** button, or
- Double click on the AMECustom icon on your desktop, or
- Type *AMECustom* in a MS DOS Command window from the directory where you wish to work (*tutorial* for instance).

The display shown in Figure 2.4: appears. It has been deliberately made similar to **AMESim** and **AMESet** but there are small differences in the display and **AMECustom** performs different functions. The main area is used to display details of submodels and supercomponents that are customized.





Note: For the rest of the tutorial exercises, you should save all your customized submodels in the same directory as the current one (*tutorial* directory).

Step 2: Select the *FX00* icon

At the left hand side of the **AMECustom** display are the *Categories* buttons. Select the *Signal, Control and Observers* category and then click on the standard f(x) icon. The dialog box below appears:



8 83	Submod	el List		? ×
	2 <mark>≯</mark> f(×	ı)≻1 fof	ĸ	
Γ	Submodel	l list		
	Name	Description		Туре
	FX00	output a press	cribed function of input	generic submodel
	E <u>x</u> ternal	variables	<u>H</u> elp	Explore
				<u>C</u> ancel

Figure 2.5: f(x) submodel list

Select the submodel *FX00* in the list and click on *OK*: Figure 2.6: shows the new display.



Figure 2.6: AMECustom display with a submodel loaded

Step 3: Customize the generic submodel

The process of customizing submodels is largely a matter of filling forms. Here you are going to alter the characteristics of the submodel *FX00* to get our own submodel *COS50*:

1. In the *Brief description* field, replace the current description by the following: 'computes the cosine of the input'. This field contains text, which will be displayed when the submodel is selected in **AMESim**.

Figure 2.7: Brief description field



2. Click on the button labeled *Full description* and change the title and description as shown in Figure 2.8:

Full Descriptio	n		?
TITLE : COS50			-
DESCRIPTION :			
COS50 ap and the re	oplies a cosine function esult is the output on po	to the input signal rt 1.	on port 2
USAGE :			-
PARAMETER SE	TTINGS :		
DATE OF CREAT	ION/AUTHOR :		
INDEX OF REVIS	IONS :		
•			<u> </u>
Clear all	Generic description	Load default	From file
		<u>0</u> K	<u>C</u> ancel

Figure 2.8: Full description

3. In the *External variables* table, modify the title of the two external variables as follows:

Figure 2.9: External variable table

External variables :			
Title	Port	Unit	Туре
cosine of input x	1	null	variable
input x	2	null	variable

4. Select the *Text parameters* tab from the *Internal items* table and set the default value of the *expression in terms of the input x* parameter to cos(x):

Figure 2.10: Internal variable table

Internal items	:		
arameters Integer parameters Text parameters			
Title		Default value	Hidden
expression in	n terms of the input x	cos(x)	

Then click on the check box of the *Hidden* column so that a tick mark appears in it:

Note: This point is important because in **AMESim**, the parameter will not appear anymore in the submodel parameter list and its default value will be used. **Hiding a customized submodel parameter ensures that its default value will always be used in AMESim.**

Step 4: Save the customized submodel

At this stage the customization is complete and you can save it by clicking on

the button. This will produce the dialog box below from which you can give a name to your customized submodel:

Figure 2.11: Submodel name and password

🛕 Enter Name and Password 🛛 🔋 🗙
Submodel name:
COS50
Save encrypted
- Password
Enter password:
Confirm password:
<u>O</u> K <u>C</u> ancel

Putting a tick mark in the check box labeled *Save encrypted* is not necessary for such a simple submodel. This feature will be covered in the last example of next chapter. If you then click on *OK*, a file browser appears and you must select an **AMESim** node directory for your customized submodel.

Figure 2.12: Directory of the submodel

Select a directory	
D:\AMETest\tutorial	
Disque local (C:)	
🖻 🚍 Disque local (D:)	
庄 🛅 ADOBEAPP	
🕀 💼 🧰 AMESetExample	
🖃 🧰 AMETest	
📄 🕀 💼 ECU_lib	
🕀 💼 FortranSubmodels	
Icons	
submodels	
tutorial	
🕀 🔁 CommonRailCompleteSystem	-

Note that if the directory you selected is not in the path list you will get the following message:

 Not In Path List
 Image: Second state s

Figure 2.13: You can add the new directory in the path list

If you click on Yes, this directory will be added to your path list.

You can get a preview of the Change Parameters and Variable List dialog

boxes you would get in **AMESim**. Click on the buttons and it to get the previews shown in Figure 2.14: and Figure 2.15: respectively:

Figure 2.14: Preview of the Change Parameters dialog box

MAMESim Paran	eter Mode	Preview	? ×
Submodel 2 ≯ f(x) ≻1	COS50 computes input	the cosine of the	External variables
- Parameters			
Title Value	Unit		
·			
<u>H</u> elp			e Options >>

Figure 2.15: Preview of the Variable List dialog box

AMESim Run Mode	e Preview	? ×
Submodel 2 ≯ f(x) ≻ 1 cr	OS50 omputes the osine of the input	External variables
Variables	Value Unit	
cosine of input x input x	null null null null	
Time:		
<u>H</u> elp		

As you can see, the parameter list is empty since the only parameter of the generic submodel is now hidden. It is set to cos(x) and cannot be changed by an **AMESim** user. The variable list shows the customized names of external variables.

Step 5: Use the customized submodel in AMESim

1. Start **AMESim** and build the small model shown in order to test your new submodel:



2. In *Submodel* mode you will get the *Submodel List* dialog box shown below after selecting the f(x) icon, and after clicking on the cross that appears to the left hand side of the list:

🍓 Submodel List		? ×						
2≯ f(x) > 1 Submodel list	fofx Currently component has no submodel.							
Name	Description	_						
⊡- <mark>F×00</mark> COS50	EX00 output a prescribed function of input COS50 computes the cosine of the input							
Copy commo	n parameters when submodel changes							
E <u>x</u> ternal variabl	es <u>H</u> elp <u>Explore</u> <u>R</u> emove	:						
	<u> </u>	el						

Figure 2.16: Submodel List dialog box

You can then select the customized submodel COS50.

If *COS50* does not appear in the list, check your pathlist and update it as necessary.

3. In *Parameter* mode, it is also possible to change between generic and customized submodels. The submodel currently assigned appears in a field and you can select another submodel from a list which is produced by clicking on the arrow in this field:

🙀 Change Paramete	15	? ×
Submodel		
2 [≯] f(x)≻1	COS50-1 computes the cosine of the input	E <u>x</u> ternal variables
Available customized s	ubmodels: COS50	_
🔽 Copy common para	meters when submu	
-Parameters	LUSOU	

Figure 2.17: List of customized submodels

Set the *slope* parameter of the ramp submodel (*RAMP0*) to 2*PI/10.

4. In *Run* mode, start a simulation for 10 seconds with a communication interval of 0.01 second and plot the output of *COS50* against its input. You should get the following curve:





As an extension to this first example, you could create other customized submodels from *FX00* such as: *SIN50*, *ABS50*, *EXP50*, etc.

2.3 Files associated with a customized submodel

Each **generic** submodel you use in **AMESim** (called *NAME* for instance) has three associated files:

- The source file *NAME.f* or *NAME.c*.
- The specification file NAME.spe.
- The compiled version of the source *NAME.o* (or *NAME.obj*).

If you edit this submodel with **AMECustom**, you can customize it according to your needs and save it as a different name: *CUSNAME* for instance. For this new submodel **AMECustom** will produce two new files which are:

- The customized specification file CUSNAME.spe.
- The customized description file CUSNAME.des.

If your customized submodel is encrypted, it is protected by a password.

2.4 Example 2: creating two customized submodels from a cam generic submodel

Objective:

• To construct more complex customized submodels.

The cam submodel *CAM00* belongs to the *Mechanical* library. The icon it is associated with is shown. This is a submodel of a cam profile with cam follower. An ASCII file is read defining the linear displacement in *mm* of the contact point at various angular displacements in *degree* in the range 0 to 360. The file containing the cam profile is defined by the user in *Parameter* mode.



A simple modification can be made to *CAM00* to produce *INCAM50* and *EXCAM50*, which will be used as inlet and exhaust cam respectively.

Step 1: Load the cam submodel into AMECustom.

At the left hand side of the **AMECustom** display are the *Categories* buttons. Select the *Mechanical* category and then click on the standard cam icon shown. The following dialog box appears:



Figure 2	2.19:	Submodel	list	dialog	box
----------	-------	----------	------	--------	-----

8	Submode	l List			? ×
	2 - D]]	- 1	cam2		
[-Submodel I	ist			
	Name	Descript	ion	Туре	
	CAM00	cam and	cam follower	generic subm	odel
	E <u>x</u> ternal v	variables	<u>H</u> elp	Explo	re
			<u> </u>	<u>C</u> a	ncel

Select the submodel *CAM00* in the list and click on *OK*: Figure 2.20: shows the new display.



AMECustom - [C:/UNNAMED_CAM00]	
3 Eile Edit Options Previews Icons Tools E	ocumentation <u>W</u> indows <u>H</u> elp
🕫 🖪 X 🖻 🖻 · 🕯	🖻 - 🛛 🛠 🔛
UNNAMED_CAM00 2 2 Change icon Brief description: cam and cam follower Full description	External items Submodel global parameters: Beal Integer Add new Add new External variables: Port Title I empve External variables: Internal items: Internal items: Internal items: Internal variables Real parameters Internal variables Real parameters Internal variables Real parameters Internal variables Inte

We are going to alter the characteristics of the submodel *CAM00* to get our own inlet cam submodel *INCAM50*. Then we will do the same for the exhaust cam submodel *EXCAM50*.

Customize CAM00 to produce INCAM50 Step 2:

- 1. In the Brief description field, replace the current description by the following: 'inlet cam and cam Brief description: follower'. This field contains text, which will be inlet cam and cam follower displayed when the submodel will be selected in AMESim.
- 2. Click on the button labeled Full description and change the title and description as shown below:

% Full Description	? ×								
TITLE : INCAM50	-								
DESCRIPTION :									
This is a submodel of an inle¢ cam profile with cam follower.									
An ASCII file is read defining the linear displacement of the contact point (in mm) according to the angular displacement (in degrees, in the range 0 to 360). Note that in this file, the displacement at the first angle value must be equal to the displacement at the last angle value.									
On port 1 the outputs are the linear speed in m/s and the displacement in m, the input is a force in N. On port 2 the output is a torque in Nm and the input is a speed in rev/min.									
There are two internal variables comprising an angular displacement state variable in degrees and a corresponding angle converted to the range 0 to 360.									
Clear all Generic description Load default From	n file								

Figure 2.21: Full description

3. In the *External variables* table, modify the title of the external variables named displacement of the cam follower and velocity of the cam follower as follows:

Figure 2.22: Modify the titles of the external variables

External variables :			
Title	Port	Unit	Туре
velocity of the inlet cam follower	1	m/s	one-line macro variable
displacement of the inlet cam tollower	1 m		multi-line macro variable

Ideally we would replace the word cam in all parameters and variables with inlet cam.

4. Select the Integer parameters tab from the Internal items table and set the default value of the parameter 1 for linear splines 2 for cubic splines to 2. Both integer parameters must be hidden:

Figure 2.23: Check boxes in the hidden column

	nternal items :								
	Internal <u>v</u> ariables	Real parameters	۱ <u>n</u>	teger parameters		Te <u>x</u> t paramet	ers		
I	Title			Minimum value	Ma	aximum value	Default	value	Hidden
I	1 for linear splines 21	for cubic splines		1		2		2	ব
I	discontinuity handling	g for linear splines: O	in	0	1	1		1	V

 Select the *Text parameters* tab from the *Internal items* table and set the default value of the parameter *file of cam position function of angular displacement* to \$*AME/tutorial/data/inlet_cam_profile.data* or %*AME*%\tutorial\data\inlet_cam_profile.data.

Note that you can use a file browser for this by clicking on the button _____, provided you know the value of the environment variable *AME*. Ensure this parameter is also hidden:

Figure 2.24: Ensure text parameter is hidden

Internal items:							
I <u>n</u> ternal variables R <u>e</u> al parar	meters	Integer parameters	Te <u>x</u> t parame	ters			
Title	Default	value		Hidden			
file of cam position function of %AME%\tutorial\data\inlet_cam_profile.data 🔽							

6. At this stage the customization of the submodel is complete and you can

save it by clicking on the *Save* button. This one will be called *INCAM50* and put in your *tutorial* directory.

Step 3: Customize CAM00 to produce EXCAM50

Repeat the procedure above (points 1 to 6) for *EXCAM50* ensuring that in points 1, 2 and 3:

- inlet cam is replaced by exhaust cam, and
- *INCAM50* is replaced by *EXCAM50*

(Alternatively with *INCAM50* still loaded in **AMECustom** do **File** \triangleright **Save as** specifying *EXCAM50* and then make the changes.)

Point 4 is identical and the result of point 5 is shown below:

Figure 2.25: Result of point 5

I	Internal items:						
ĺ	l <u>n</u> ternal variables	R <u>e</u> al para	ameters	Integer parameters	Te <u>x</u> t parameters	:	
	Title		Default v	value		Hidden	
	file of cam position function of %AME%\tutorial\data\exhaust_cam_profile.data 🗹						

Step 4: Test your customized submodels in AMESim

1. Start **AMESim** and build the model shown in Figure 2.26: in order to test your new submodels. You might get *LCON12* instead of *LCON11*

depending on use of 'flip' operations but this is not important. Save this model in your *tutorial* directory.



Figure 2.26: Build this model

2. Use the following parameters for the inlet and exhaust side:

Submodel	Title	Value
SPR000A	SPR000A spring force with both displacements zero [N]	
MAS005	mass [kg]	0.01
	lower displacement limit [m]	0
	higher displacement limit [m]	0.03
LSTP00A	gap or clearance with both displacements zero [mm]	0.1
	contact stiffness [N/m]	1.0e9
	contact damping [N/(m/s)]	
	penetration for full damping [mm]	0.01

3. Run a simulation for 0.08 second with a communication interval of 0.0001 second and plot the two cam displacements against their angular displacement. You should get the following curves:



Figure 2.27: Cam displacements against their angular displacement

4. Save this model carefully because it is used again in next chapter.

Chapter 2 Customizing submodels

Chapter 3: Customizing supercomponents

AMECustom allows you to customize supercomponents as well as submodels. In this chapter there are two examples.

In the first you construct a system which models a cam using mechanical and control library components. You then convert it to a supercomponent. This produces the same answers as the standard submodel *CAM00*. However, it is not very convenient to use. To overcome this you customize it so as to make it appear very similar to *CAM00*.

In the second example you use a customized cam in a supercomponent of an inlet and exhaust value of a cylinder head. You then customize this to have two levels of customization.

It is useful but not essential that you have some experience of using the *Icon designer* in **AMESim** and/or **AMESet**.

3.1 Example 1: A customized cam supercomponent

Objectives of the example:

- To illustrate the process of customizing a supercomponent.
- To compare a generic submodel, generic supercomponent and customized supercomponent of the same component.

Figure 3.28 shows a system using the standard cam submodel CAM00.

Figure 3.28: System using the cam submodel CAM00

SUBMODEL CAMOO мот

Figure 3.29 shows the external variables of the cam submodel CAM00.

Figure 3.29: External variables of the submodel CAM00



It is possible to produce the same results with the cam submodel replaced by mechanical and control components. Figure 3.30 shows how this can be done.







We describe this as a 'flat system' because it is single level and will soon be transformed into a supercomponent. A few words of explanation are needed.

Ignoring units by assuming everything is SI the logic is relatively simple. The rotary speed ω is integrated to produce the cam position θ .

$$\theta = \int \omega dt$$

This is used with a table of values imported from a file to produce a displacement x and the derivative $dx/d\theta$. From this we can recover the velocity v as follows:

$$v = \frac{\delta x}{\delta t} = \frac{\delta x}{\delta \theta \delta t} = \frac{\delta x}{\delta \theta} \omega$$

For a perfect transformer we have

$$Fv = F\frac{\delta x}{\delta \theta}\omega = T\omega$$

From which

$$T = F \frac{\delta x}{\delta \theta}$$

The four gains are needed to produce the correct units. Thus the table of values used in *CAM00* and *FXA10* give a cam lift in *mm* for angles in *degree*. This is inconvenient from a modelling point of view but this is how data of this type is normally presented!

3.1.1 Constructing the flat system

Step 1: Construct both models in the same system

Submodel	Reference number on sketch if any	Title [units]	Value
GA00	1	value of gain [null]	6
GA00	2 and 3	value of gain [null]	0.001
GA00	4	value of gain [null]	180/pi
PM000	5 and 6	shaft speed [rev/min]	150
CAM00		file of cam position function of angular displacement	\$AME/tutorial/data/cam.data
		1 for linear splines 2 for cubic splines	2
FXA010		name of ASCII file	\$AME/tutorial/data/cam.data

Step 2: Set the following parameters leaving all others at their default values

Step 3: Run a simulation and check that the results from the two models are the same

Use a final time of 1 second and enable discontinuity printout.

Figure 3.31: Plot of the CAM00 and VELXC0 submodels displacement



1 - CAM00-1 displacement of the cam follower [m]

3.1.2 Constructing the generic supercomponent

Step 1: Convert the sub-system shown in Figure 3.32 to a generic supercomponent using the cam icon



Figure 3.32: Sub-system to be converted to a generic supercomponent

Figure 3.33: Specify an icon, a name and a brief description



Step 2: Add this to your AMESim system and check that it gives the same results as the other two versions



Comparing the generic supercomponent with the generic submodel we have to prefer the submodel. There are far too many redundant parameters in the supercomponent which should be left at the set values. If we released the supercomponent for others to use, sooner or later some of these values will get changed. The situation is unsatisfactory so we will see what **AMECustom** can do to improve matters.

The basic idea is to get the supercomponent to behave more like the submodel *CAM00* hiding irrelevant detail. In particular we want the *Change Parameters* and *Variable List* dialog boxes to be like those in Figure 3.35 and Figure 3.36 respectively.

Figure 3.35: Change	Parameters dialog box

Submodel			
2511-1	CAM00-1		External variables
	Cam and cam follower		
Parameters			
Title		Value	Unit
angular displacem	ent of the cam		0 degree
1 for linear splines	2 for cubic splines		1
discontinuity hand	lling for linear splines: 0 inactive 1 active		1
file of cam positior	n function of angular displacement	P:/AMESim/v4Beta/tutorial/data	ı/cam.data

Submodel				
2 91 -1	CAM00-1 Cam and cam follower		E <u>x</u> ternal variables	
Variables				
Title		Value	Unit	Saved
velocity of the car	velocity of the cam follower		0 m/s	ব
displacement of th	displacement of the cam follower		Om	N
force on the cam I	force on the cam follower		1 N	N
torque on cam	torque on cam		0 Nm	V
rotary velocity of the cam		15	0 rev/min	V
angular displacement of the cam		90	0 degree	V
angular displacement of the cam modulo(360)		18	0 degree	N
transformer ratio			Om	

Figure 3.36: Variable List dialog box

3.1.3 Customizing the supercomponent

Step 1: Load the supercomponent into AMECustom. Try the following experiments:

1. Open the supercomponent tree structure as in Figure 3.37.

Figure 3.37: Supercomponent tree structure

2=					
Change icon					
Brief description:					
generic supercomponent ca	m				
Eull description					
Submodel	T				
Joubillouer	liype				
E GENSUPERCAM5) supercomponent				
GENSUPERCAM5	U supercomponent component				
E-∞ GENSUPERCAM5 	D supercomponent component component				
É- ↔ GENSUPERCAM5 	D supercomponent component component component				
É- ↔ GENSUPERCAM5 	D supercomponent component component component component				
É- ↔ GENSUPERCAM5 	D supercomponent component component component component component				
É- ↔ GENSUPERCAM5 	D supercomponent component component component component component component				
É- ↔ GENSUPERCAM5 * FT010 	D supercomponent component component component component component component component				
Gubinous Gensupercam5 Gensupercam5 Gaoo Gaoo Gaoo Gaoo Gaoo Gaoo Gaoo Gaoo Gaoo Gaoo Gaoo Gaoo Gaoo Gaoo	D supercomponent component component component component component component component component				
GENSUPERCAM5 GENSUPERCAM5 GENSUPERCAM5 GENSUPERCAM5 GA00 GA0	D supercomponent component component component component component component component component				

2. Click on the *Explore* button to produce the *Explore Supercomponent* dialog box part of which is shown in Figure 3.38. Click on individual constituents of the supercomponent and note how the corresponding item gets selected in the supercomponent tree.



Figure 3.38: Constituents of the supercomponent

- 3. Note that as you select constituents of the supercomponent, on the left side the two upper lists on the right are unchanged but the lowest list changes.
- 4. Do a *Parameter* mode and *Run* mode preview and note how messy it is! In particular notice how prefixes, composed from the submodel name and instance number, are added to some parameter/variable titles in order to make them unique.

Figure 3.39: Parameter and variable titles

GA00-1: value of gain	6 null
INT0-1: value of gain	1 null
offset to be added to input	0 null
minimum output value	0 null
maximum ouput value	360 null
GA00-2: value of gain	0.001 null
GA00-3: value of gain	0.001 null
GA00-4: value of gain	180/piinull

Normally it is confusing to see these prefixes in a *Change Parameters* or *Variable list* dialog box. They can be removed in 2 ways:

- By hiding the parameter or variable so that it does not appear anymore.
- By changing the title. If you do this, it becomes your responsibility to ensure parameters and variables have unique titles.

Step 2: The central list is what will become the external variables of the customized supercomponent. Alter the titles as shown below:

Figure 3.40: Alter the titles of external variables



Step 3: Select the root of the supercomponent tree and right-click selecting Hide all

Figure 3.4 ⁴	I: Hide all	in the su	percomponen	t tree
-------------------------	-------------	-----------	-------------	--------



You can achieve the same thing using the right-click pulldown menu in the *Explore supercomponent* dialog box.

Figure 3.42: Hide all using the Explore supercomponent dialog box

Component variables and parameters 🔸	Hide all
Labels •	Show all
Bird's eye view	Copy parameters
i ! 🖓	Paste parameters
	Image: Second secon

If you look at the *Parameter* and *Run* mode previews, you will see a dramatic change! We must reintroduce a few variables and parameters that have been hidden.
Γ!	Parame	eters —							 Ì
	Title	Value	Unit						
'									
									 Ī
		– Variable	es —— es						
		Title				Value	Unit		
		velocity	y at por	t3		nul	lm/s		
		displac	ement p	port 3		nul	lm		
		force a	it port 3			nul	IN		
	output torque port 3			nullNm					
		input a	ngular v	/elocity	at port 3	nul	l rev/min		

Figure 3.43: Some variables and parameters are hidden

Below are summarized three very important points:

- 1. It is very common practice to begin by hiding ALL parameters and variables then progressively reintroducing those that are needed.
- 2. After making any significant change, it is essential to do a *Parameter* and *Run* mode preview. This helps you to identify mistakes at an early stage.
- 3. You can change titles of parameters and variables to make them more meaningful and to remove unwanted prefixes. If you do this, it becomes your responsibility to ensure the titles are unique.

Step 4: Select the integrator submodel *INT0*, then select the *Internal variables* tab and alter them as follows:

I	Internal items:									
	l <u>n</u> ternal varial	bles R <u>e</u> al	paramete	rs	Integer	parameter:	s Te <u>x</u> t	parameter	rs	
	Title		Unit	Ty	ре	Minimum	Default	Maximum	Hidden	
	INT0-1: output	t from integral	tor null	exp	olicit st	-1e+030	0	1e+030		
nternal items:										
Internal variables Real parameters Integer parameters Text parameters										
Title			L	Init	Туре		Minimum	Default	Maximum	Hidden
angular displacement of cam in degree. null explicit state vari1e+030 0 1e+030										

Figure 3.44: Alter the Internal variables

Note the Hidden flag for the variable has been changed, now it is unticked.

Step 5: In a similar way make the following changes. In each case ensure the *Hidden* check box is not ticked.

Chapter 3 Customizing supercomponents

Submodel	Tab selected	Original title	New title if different
MOD00	Internal variables	output	angular displacement of cam modulo 360 in degree
FXA010	Integer parameters	discontinuity handling	
FXA010	Text parameters	name of ASCII file	file of cam position function of angular displacement

Do a preview of parameters (Figure 3.45) and variables (Figure 3.46) to check they are satisfactory.

Figure 3.45: Preview of parameters

- F	Parameters			
	Title	Value	Unit	[
l	# angular displacement of cam in degree	0	Inull	
l	discontinuity handling	inactive)	
l	file of cam position function of angular displacement	\$AME/tutorial/data/cam.data	I	

Figure 3.46: Preview of variables

-1	/ariables		
	Title	Value	Unit
	velocity of cam follower	nul	lm/s
	displacement of cam follower	nul	lm
	force on cam follower	nul	IN
	torque on cam	nul	INm
	rotary velocity of cam	nul	lrev/min
	angular displacement of cam in degree	nul	Inull
	angular displacement of cam modulo 360 in degre	nul	Inull

Step 6: Save the customized supercomponent under the name *CUSTOMCAM50*.

Step 7: Use the customized supercomponent in **AMESim** adding to the three existing subsystems. Check it produces the same results as the other versions. Examine the *State count* dialog box to see how certain titles are hidden.

Figure 3.47: Supercomponent used in an AMESim system



Conclusion

This customized supercomponent is an alternative to the submodel *CAM00*. Which alternative is better? In this case the submodel is less complex and has less redundancy. In other cases a customized supercomponent can be a better alternative to a huge submodel. This is probably true in the next example.

3.2 Example 2: A 2-level customized supercomponent with global parameters

3.2.1 Introduction



Figure 3.48: Two valves operating within a cylinder head

Objectives:

- To construct a customized supercomponent with two levels of customization;
- To use parameters global to the supercomponent;
- To demonstrate encryption facilities.

The system is shown in Figure 3.48 and represents two valves operating within a cylinder head of an engine. The exercise that follows is constructed to show features of **AMECustom** not to serve as instruction on how to design efficient cylinder head valve systems!

Note that the system is precisely the same as the second example in chapter 2. Remember that:

- There are two camshafts running at the same speed which is defined by *PM000*.
- *INCAM50* and *EXCAM50* are customized submodels. This will give two levels of customization.
- The submodels *LSTP00A* induce a valve clearance when the valve is closed.
- MAS005 features a limit in the movement of the valves.

3.2.2 Constructing the flat system

Reload the old system and check the results against Figure 3.49. This is the valve lift and the reference set of results we will use to check the new generic supercomponent and its customized version.



Figure 3.49: Reconstructing the generic supercomponent

Step 1: Create a generic supercomponent including all components except *PM000*.

Auxiliary System		?×
	Supercomponent creation	
	Icon: No icon specified	Select an jcon
콜콜	Name:	
ا محم وحم ا	Brief description:	
╽╷╌╌╴╵╷╧╵╵╼╷╺┛╽		Eull description
		<u>S</u> ave
	<u></u>	
<u>Flip</u> <u>R</u> otate	Paste Print Supercomponent	Close

Figure 3.50: Create the supercomponent

There is no very suitable standard **AMESim** icon for this supercomponent. If you are very short of time you can use any icon of the *Mechanical* category

мть

which has a single rotary port such as the following one:

However, for a more artistic solution we suggest you do the following:

- 1. Click on Select an icon.
- 2. You should select a category for which you have write permission. If you do not have one, you must create it by clicking on the *New Category* button.
- 3. Click on New Comp Icon.
- 4. Click on the *Load icon file* button of the *Icon designer* dialog box.
- 5. There is an existing icon in a folder of the **AMESim** system area. Enter the following name:

\$AME/tutorial/Icons/twinvalves.xbm

or

%AME%\tutorial\Icons\twinvalves.xbm

6. The icon should now be loaded into the *Icon designer*. Define a rotary port, an icon name and a brief description.

🖬 Icon designer	? ×
🗋 😂 🖬 🛃 🖍 🌐 Zoom (%): 400 💌 Width: 66	井 Height: 🌆 🛓
	Ports on icon: 66 7 rshaft
Icon name: Itwinvalves Brief icon description:	twin valves in a cylinder head
Cursor coordinate: 66 , 7	Cancel

Figure 3.51: Draw the icon

Step 2: Save the icon.

Make sure you do this in **AMESim** format

Step 3: Select the icon you have just created from the tree structure of the *Icon selection* dialog box.

- 1. Give a name the generic supercomponent (e.g. VALVES50),
- 2. Give it a brief description,
- 3. Save it.

Figure 3.52: Give a name and brief description

- Supercompo	nent creation	
lcon		Select an icon
Name	VALVES50	
Brief description	twin valves in a cylinder head	
·		Full description
		Save

Step 4: Add the generic supercomponent to your **AMESim** system and run a simulation to check that the results are correct (as in Figure 3.49).

Figure 3.53: Add the supercomponent to the AMESim system



3.2.3 Constructing the customized supercomponent

- Step 1: Load the generic supercomponent into AMECustom. Hide all variables and parameters.
- Step 2: The original external variable names shown in Figure 3.54 have very inappropriate names. Change them to those indicated.



For the time being, only external variables of the supercomponent are visible. We must decide on what else is to be visible. There are many pairs of submodels in this supercomponent. In order to demonstrate the use of global parameters, we will assign the same values to many parameters from the intake side and from the exhaust side. This symmetry is of course not necessary and may not be desirable from a design point of view, but this is not the point of the exercise.

It would be convenient to have the following parameters to set in **AMESim**:

- valve spring preload
- valve spring stiffness
- valve clearance
- valve mass
- maximum possible valve lift

We set these parameters once and they are applied to both the intake and the exhaust side. To do this we introduce customized supercomponent global parameters in **AMECustom**. These are like global parameters in **AMESim** but their scope is solely the customized supercomponent.

The table below shows the global parameters we will use and an indication of where they will be used. They are all *Real parameters* but we could also use *Integer* and *Text parameters*.

Global type and name	Title [units]	Min./Default/ Max.	Submodel, parameter/variable and title
Real VPRELOAD	valve spring preload [N]	200/300/800	<i>SPR000A</i> , real parameter, free length of spring
Real VSTIFF	valve spring stiffness [N/m]	1.0e4/1.0e5/ 1.0e6	<i>SPR000A</i> , variable, spring rate
Real VCLEAR	valve clearance [mm]	0.05/0.1/0.2	<i>LSTP00A</i> , variable, gap or clearance with both displacements zero
Real VMASS	valve mass [kg]	0.005/0.01/ 0.05	<i>MAS005</i> , real parameter, mass
Real VMAXLIFT	maximum possible valve lift [m]	0.02/0.03/ 0.085	<i>MAS005,</i> real parameter, higher displacement limit

Step 3: Introduce the supercomponent real global parameters shown below. Make sure that none of them are hidden.

	upercomponent global parameters:								
<u>B</u> eal Integer Iext									
	Name	Title	Unit	Minimu	Default v	Maximum v Hidd			
	VPRELOAD	valve spring preload	N	200	300	800			
	VSTIFF	valve spring stiffness	N/m	1.0e4	1.0e5	1.0e6 🗖			
	VCLEAR	valve clearance	m	0.5e-4	1.0e-4	2.0e-4 🗖			
	VMASS	valve mass	kg	0.005	0.01	0.05			
	VMAXLIFT	maximum possible valve lift	m	0.02	0.03	0.85 🗖			

Figure 3.55: Alter the external variable names

Note that you click on the *Add new* button to introduce each new real parameter.

Step 4: Use the *VPRELOAD* and *VSTIFF* global variables in both *SPR000A* spring submodels *Real parameters* as shown below.

l	Internal items:							
I	l <u>n</u> ternal variables	R <u>e</u> al parameters	Integer para	meters	Te <u>x</u> t para	ameters		
I	Title SPR000A-1: spring rate SPR000A-1: spring force with both displacements zero			Unit	Minimum v	Default valu	e Maximum	Hidden
I				N/m	0	VSTIF	F 1e+008	
l				N	-1e+006	VPRELOAI) 1e+008) V

Figure 3.56: Global variables

Note that you can copy/paste from one spring to the other using the tree structure.

Figure 3.57: Copy/paste	using	the tree structure

mass_medior	n_enusiops M	4			
spring01	Hide all	ernal items:			
zeroforcesc	Show all	<u>i</u> ternal vari			
zeroforcesc	Copy parameters	itle			
zerospeeds	Paste parameters	PR000A-1:		SI	al itema:
2eiospeeus		SPRUUUA-1:	spring01	Hide all	aritems.
			zeroforcesource	Show all	al varia
			···· zerotorcesource	Convinarameters	04.2
			zerospeedsourci	Paoto parametero	UA-2:
			olacticondator	r aste parameters	OA-2.

You can do the same thing in the Explore Supercomponent window.

Figure 3.58: Copy/paste using the Explore Supercomponent window

Ť,		_		
	Component variables and parameters		ŀ	lide all
Ji	External variables		9	ihow all
	Labels 🕨		C	Copy parameters
	Bird's eye view		F	aste parameters
	Help			
μ.				
2	Component variables and parameters	Г '	, -	, lide all
	Component variables and parameters	Г	, + 9	, lide all ihow all
	Component variables and parameters	-	, + 9	iide all ihow all copy parameters
	Component variables and parameters External variables Labels Bird's eye view		، ۲ ۵	lide all ihow all Copy parameters Paste parameters

A third technique is to select **both** springs in the tree structure or both springs in the *Explore supercomponent* window. This works like *Common Parameters* in **AMESim** and you can set the values for both springs simultaneously.



Figure 3.59: spring values can be set simultaneously

Step 5: Using the same techniques set the *VCLEAR* global parameter in both LSTP00A submodels as indicated.

Figure 3.60: VCLEAR global parameter

h	nternal items:								
ĺ	l <u>n</u> ternal variables	R <u>e</u> al parameters	Integer parameters		Te <u>x</u> t	: parameter:	;		
	Title			Uni	it	Minimum v	Default va	Maximum va	Hic
	LSTP00A-1: gap or (clearance with both d	lisplacements zero	mm		-0.001	VCLEAR	1000	N
	LSTP00A-1: contact	t stiffness		N/m	1	1e-020	1e+009	1e+030	

Step 6: For both *MAS005* submodels make the modifications shown below.

Internal items:	
Internal variables Real parameters Integer par	meters Te <u>x</u> t parameters
Title	Unit Minimum V Default valu Maximum v Hid
MAS005-1: mass	kg 1e-006 VMASS 1e+006√
MAS005-1: coefficient of viscous friction	N/(m/s) 0 0 1000 🔽
MAS005-1: coefficient of windage	N/(m/s)**2 0 0 1000 🔽
MAS005-1: Coulomb friction force	N 0 0 10000 🔽
MAS005-1: stiction force	N 0 0 10000 🔽
MAS005-1: lower displacement limit	m -1e+030 0 1e+030 🔽
MAS005-1: higher displacement limit	m -1e+030 VMAXLIFT 1e+030 🗹
MAS005-1: inclination (+90 port 1 lowest, -90 port 1 hig	nest) degree 🛛 -90 🔽 🛛

Step 7: In the inlet *MAS005* submodel alter the Internal variables as shown. Make the corresponding changes in the exhaust side.

	Internal items:								
	l <u>n</u> ternal variables	R <u>e</u> al parameters	Integer p	paramete	ers	Te <u>x</u> t parameters			
I	Title			Unit	Тур	e	Minimum	Default	Maximum Hid
	MAS005-1: activity o	f mechanical dissipa	tion (frict)	J	activ	/ity variable			V
I	MAS005-1: activity o	f mechanical inertia	(mass)	J	activ	/ity variable			V
	inlet valve velocity			m/s	expl	icit state variable	0	0	0
	inlet valve lift			m	expl	icit state variable	0	0	0
	inlet valve accelerati	on		m/s/s	varia	able			

- **Note:** 3 (x2) *Internal variables* have had their titles altered and are not hidden anymore.
 - 2 (x2) *Internal variables* have their minimum, maximum and default values set to 0. This makes them invisible in *Parameter* mode but as they are no longer hidden, you will see them with their new titles in *Run* mode.

Step 8: In the inlet cam *INCAM50* submodel alter the second Internal variable as shown. Do not change the exhaust side.

Internal items:								
l <u>n</u> ternal variables	Real parameters	Integer	paramete	ars ∐ Te <u>x</u> tp	paramete	ers		
Title			Unit	Туре	Minimu	Defa	Maximu	Hide
INCAM50-1: angular	displacement of the	cam	degree	explicit st	0	0	360	
angular positions of c	cams		degree	multi-line				
INCAM50-1: transform	mer ratio		m	multi-line				$\mathbf{\nabla}$
velocity of the inlet c	am follower		m/s	one-line				\checkmark
displacement of the in	nlet cam follower		m	multi-line				~
torque on inlet cam			Nm	one-line				~

Figure 3.63: Uncheck the Hidden box for the second variable

- **Note:** We have forced both cams to have a starting angle of 0. We will not be able to change this.
 - We have forced the starting value of the *gap or clearance* in both *LSTP00A* submodels to be *VCLEAR*.
 - With both cam profiles starting on a flat part the starting values of various state variables are consistent.

Step 9: Do a Parameter mode preview (Figure 3.64) and a Run mode preview (Figure 3.65).

N 🏷	MESim Parameter Mod	de F	Preview				<u>?</u> ×
-S	iubmodel						
		NA 1 Val	MED_V#	ALVES ylinder	50-1 	<u>s</u> ternal va	riables
F	arameters		Value	Lloit			
	valve spring preload valve spring stiffness valve clearance valve mass maximum possible valve	lift	300 100000 0.0001 0.01 0.03	N/m M kg m			
	<u>H</u> elp				<u>C</u> lose	0ptic	ons >>

Figure 3.64: Parameter mode preview

Figure 3.65: Run mode preview

🖄 AMESim Run Mode I	Previe	w			<u>? ×</u>
	UNN twin v cylind	AMED_ valves in ler head	VALVE S a	50-1	E <u>x</u> ternal variables
Variables Title		Value	Unit		
cam torque rotary velocity of cam angular positions of ca inlet valve velocity inlet valve lift inlet valve acceleratio exhaust valve velocity exhaust valve lift exhaust valve acceler	ams n , ration	nul nul nul nul nul nul nul	INm Irev/min Idegree Im/s Im/s/s Im/s Im/s Im		
Help					

- Step 10:Save the customized supercomponent as *CUSVALVES50*. This time specify a password, but do not forget it!
- Step 11:Add the customized supercomponent to your **AMESim** sketch and check that it produces the same results.



Figure 3.66: Add the customized supercomponent to the sketch

We hope that you find using *CUSVALVES50* within **AMESim** more pleasant than the generic equivalent.

Step 12:Use the state count facility and also examine in an editor the *.state* and *.var* files produced by AMESim.

The important thing to realize is that things hidden by **AMECustom** appear under very unhelpful names (Figure 3.67). This is true even if it is not encrypted. In **AMECustom** you define what is private and what is public. **AMESim** only shows the public part.

State NO		Submoder	vanable	
1	331		hidden variable	
2	U		hidden variable	
3	77724	CUSVALVES50-1	inlet valve velocity	m/s
4	0	CUSVALVES50-1	inlet valve lift	m
5	53466	CUSVALVES50-1	exhaust valve velocity	m/s
Б	1	CUSVALVES50-1	exhaust valve lift	m
ïme:0.1 s pdate		utomatic update		

Figure 3.67: State count dialog box

3.2.4 Activity variables and customized supercomponents

Following the steps in the previous section the status of all activity variable are hidden. For this reason you will not have access to these variables in **AMESim** regardless of the state of the *Activity index calculations* check box in the *Run Parameters* dialog box.

We will now reintroduce these activity variables.

1. In **AMECustom** uncheck the hidden box for all activity variables. You will then get the following *Run* mode preview.

Variables		
Title	Value	Unit
cam torque	nul	INm
rotary velocity of cam	nul	l rev/min
angular position of cams	nul	ldegree
LSTP00A-1: activity of mechanical dissipation (damp)	nul	IJ
LSTP00A-1: activity of mechanical capacitance (spring)	nul	IJ
LSTP00A-2: activity of mechanical dissipation (damp)	nul	IJ
LSTP00A-2: activity of mechanical capacitance (spring)	nul	IJ
inlet valve velocity	nul	lm/s
inlet valve lift	nul	lm
inlet valve acceleration	nul	lm/s/s
MAS005-1: activity of mechanical dissipation (frict)	nul	IJ
MAS005-1: activity of mechanical inertia (mass)	nul	IJ
SPR000A-1: activity of mechanical capacitance (spring)	nul	IJ
SPR000A-2: activity of mechanical capacitance (spring)	nul	IJ
exhaust valve velocity	nul	lm/s
exhaust valve lift	nul	lm
exhaust valve acceleration	nul	lm/s/s
MAS005-2: activity of mechanical dissipation (frict)	nul	IJ
MAS005-2: activity of mechanical inertia (mass)	nul	IJ

Figure 3.68: Run mode preview

Actually since there is no friction in the *MAS005* submodels and no possibility of adding friction in **AMESim**, these two particular activities will always be zero and hence can be hidden.

2. Take each activity variable in turn and remove the unwanted prefix and adjust the titles to make them unique. Below are some suggestions.

Activity index calculations

I	\Box	/ariables		
I		Title	Value	Unit
I		cam torque	null	Nm
I		rotary velocity of cam	null	rev/min
I		angular position of cams	null	degree
I		activity of mechanical dissipation (inlet constact damping)	null	J
I		activity of mechanical capacitance (inlet contact spring)	null	J
I		activity of mechanical dissipation (exhaust contact damping)	null	J
I		activity of mechanical capacitance (exhaust contact spring)	null	J
I		inlet valve velocity	null	m/s
I		inlet valve lift	null	m
I		inlet valve acceleration	null	m/s/s
I		activity of mechanical inertia (inlet mass)	null	J
I		activity of mechanical capacitance (inlet valve spring)	null	J
I		activity of mechanical capacitance (exhaust valve spring)	null	J
I		exhaust valve velocity	null	m/s
I		exhaust valve lift	null	m
I		exhaust valve acceleration	null	m/s/s
		activity of mechanical inertia (exhaust mass)	null	J

Figure 3.69: You can remove unwanted prefixes

- 3. Save the customized supercomponent and load the revised version into **AMESim**.
- 4. Enable Activity index calculations and do a run.
- 5. Use **Tools** Activity index to produce an *Activity Index List* dialog box.

Figure 3.70: Activity Index List dialog box

Submodel	Title	Value	Unit	Туре	Index	%
CUSVALVES50-1	activity of mechanical dissipation (exhaust contact damping)	1.966e-004	J	R	1.467e-004	%
CUSVALVES50-1	activity of mechanical dissipation (inlet constact damping)	3.962e-004	J	R	2.955e-004	%
CUSVALVES50-1	activity of mechanical capacitance (inlet contact spring)	2.791e-002	J	С	2.082e-002	%
CUSVALVES50-1	activity of mechanical capacitance (exhaust contact spring)	3.007e-002	J	С	2.243e-002	%
CUSVALVES50-1	activity of mechanical inertia (inlet mass)	1.059e+000	J	1	7.896e-001	%
CUSVALVES50-1	activity of mechanical inertia (exhaust mass)	1.444e+000	J	1	1.077e+000	%
CUSVALVES50-1	activity of mechanical capacitance (inlet valve spring)	6.474e+001	J	С	4.829e+001	%
CUSVALVES50-1	activity of mechanical capacitance (exhaust valve spring)	6.676e+001	J	С	4.980e+001	%

3.3 Conclusion

You should now have some idea of what **AMECustom** does. At this point it is appropriate to reread section 1.5 where ideas on how **AMECustom** can be used are presented.

Chapter 3 Customizing supercomponents

Chapter 4: Reference Guide for AMECustom

4.1 Introduction

This guide is designed to be used for reference from the main index. It is organized as follows:

- a description of The AMECustom main window,
- features available from The AMECustom menu bar,
- features available from The AMECustom Toolbar,
- categories available from The Category buttons,
- a description of The customized object general features,
- a presentation of The global parameter lists,
- a presentation of The External variables list,
- a presentation of The Internal items lists.

4.2 The AMECustom main window

The empty main window of **AMECustom** is shown in Figure 4.71. Click on the menu bar or move your mouse pointer on the toolbar and the category buttons to have more details:



Figure 4.71: AMECustom display

Category buttons

When an object is loaded, the main window looks like in Figure 4.72.





Customized object general features

This nomenclature will be used throughout this chapter.

Note: If a customized submodel is loaded rather than a customized supercomponent, the customized object general feature area is simpler than the one shown in Figure 4.72.

4.3 The AMECustom menu bar

Each menu allows you to access the main **AMECustom** commands. See the details in the following sections:

- Icons menu
- Documentation menu

4.3.1 File menu

<u>F</u> ile	<u>E</u> dit	<u>O</u> ptions	<u>P</u> reviews	lco <u>n</u> s	<u>T</u> ools	Documentation	Windows
6	<u>0</u> pe	n				C	trl+O
	<u>S</u> av	е				C	trl+S
	Sav	e <u>a</u> s					
	1: D	:/ModelsF	orManual/:	submod	els/INC	AM50.spe	
	2: D	:/ModelsF	orManual/:	submod	els/VAL	VES50.spe	
	3: D	:/ModelsF	orManual/:	submod	els/CUS	STOMCAM51.spe	
	4: P:	/AMESim	/v4.2.0/sul	bmodels	MCLC	0AA.spe	
	<u>C</u> los	е					
	<u>Q</u> uit					C	trl+Q

Figure 4.73: File menu

Open

This method allows you to select generic or customized objects by name rather than by icon.

To open an object you can also:

- click on the button in the Toolbar,
- use the Ctrl+O shortcut.

If you select *Open* in the *File* menu, you will get a file browser to enable you to look for *.spe* files. Hence you should normally search in the *submodels* folder of **AMESim** nodes.

Open					? ×
Look jn:	C AMECustom	•	E	<u>e</u> ř	
submode	ls				
File <u>n</u> ame:	[<u>O</u> pen
Files of type:	Submodel specification files (*.spe)		-		Cancel

Figure 4.74: File browser in the Open dialog box

You can open .spe files of:

- · generic submodels,
- generic supercomponents,
- · customized submodels,
- customized supercomponents.

Any object selected that can be read will be loaded into the **AMECustom** main window.

Note that all successful save operations in **AMECustom** will always produce **customized** objects.

Save

Select this menu item when you think your customization is complete and you want to save it. If the customized object is new, *Save* behaves like *Save as*.

You can also click on the *Save* button in the *Toolbar* or use the **Ctrl+S** shortcut.

Save as

Select this menu item when you want to save a customized object under a new name and/or in a different **AMESim** node. You must specify the name and also the **AMESim** node (Figure 4.75).

🖪 Enter Name and Password 🔋 🗙	Browse for Folder	? ×
Submodel name: CUSTOMCAM	Select a directory C:\AMETest	
Save encrypted Password Enter password: Confirm password:	AMETESL	×
	ОК	ancel

Figure 4.75: Save as process

Note: You can save a customized object encrypted. If you do this, you will have to set a password. Encrypted objects cannot be opened in **AMECustom** without their password.

Last opened files list

This section of the menu displays the last opened submodels or supercomponents.

You can change the number of the last opened files to display by using **Options** AMECustom Preferences...

Close

Select this item when you wish to clear the active submodel or supercomponent.

If **AMECustom** thinks it has changed, you will be invited to save it.

Quit

Select this item when you wish to leave **AMECustom**. If loaded objects have changed, you will be invited to save them.

You can use the Ctrl+Q shortcut.

4.3.2 Edit menu

The first eight items only operate on a selected global parameter.

Chapter 4 Reference Guide for AMECustom



Figure 4.76: Edit menu

Cut

Х

The selected global parameter is removed. The shortcut is **Ctrl+X**. The facility is also available by clicking on the *Cut global parameter* button in the toolbar.

Сору



The selected global parameter is copied into a special buffer or clipboard. The shortcut is **Ctrl+C**. The facility is also available by clicking on the *Copy current item* button in the toolbar.

Paste



The contents of the selected global parameter are overwritten with the contents of the clipboard. The shortcut is **Ctrl+V**. The facility is also available by clicking on the *Paste current item* button in the toolbar.

Delete

The selected global parameter is deleted. The shortcut is Del.

Add new

A new global parameter is appended at the end of the list. The shortcut is Ins.

Trash can

	1
	Þ.
0000	

This opens a dialog box from which the latest global parameters that have been deleted are shown:

é	Trash Can						? ×
	<u>R</u> eal params	<u>I</u> nt.	params	<u>T</u> ext pa	arams		
	Source submod	el	Title		Unit	Name	Value
	CUSVALVES50		valve clea	arance	m	VCLEAR	0.0001
	CUSVALVES50		valve ma:	88	kg	VMASS	0.01
	Nb elements: 2			Re <u>s</u> l	tore	Empty	Trash Can

It is then possible to restore some of these or to empty the trash can. The shortcut is **Ctrl+T**.

Move up

If it is possible, the selected global parameter is moved up one position in the list. The shortcut is **Ctrl+Up**.

Move down

If it is possible, the selected global parameter is moved down one position in the list. The shortcut is **Ctrl+Down**.

Hide all

This option hides all internal items of the customized object currently selected: a tick mark appears in every check box of the *Hidden* column.

Show all

This option shows all the internal items of the customized object currently selected: all the tick marks in the check boxes of the *Hidden* column are removed when possible.

Update categories

This option updates the categories and their contents according to the current path list.

External variables

This option produces a dialog box showing the external variables associated with each port of the selected customized object.

Figure 4.77: External Variables window

🔏 External Variables 🛛 💡	×
INCAM50	
inlet cam and cam follower	
For variables which have a direction associated with them, a positive sign is in the direction of the arrow.	
$ \begin{array}{c} \leftarrow Nm \\ \rightarrow rev/min \end{array} 2 $	

Available customized

This option produces the list of available customized objects (submodels or supercomponents) from the current path list. You can select from this list and load into **AMECustom** for editing (click on *Load*) or deleting (click on *Remove*) any existing customized object.

Figure 4.78: Available Customized Submodels dialog box

Name	Туре	Icon		Load
ÈD:/AMETest/tutorial/submodels	Storage directory			
cam2	Component icon	-JP-		
CUSTOMCAM50	Customized component			
INCAM50	Customized component			
EXCAM50	Customized component		-	<u>B</u> emove

Available supercomponents

This option produces the list of both generic and customized supercomponents that are available from the current path list. You can select from this list and load into **AMESim** (click on *Load*) or load into **AMECustom** (click on *Customize*) or remove (click on *Remove*) any existing supercomponent.

Name	Type Icon <u>L</u> oad
⊕- rotaryload2	Component icon
⊡- pump03	Component icon
PUMP1CYL	Supercomponent
4	▶ <u>R</u> emove

Figure 4.79: Available Supercomponents dialog box

Available user submodels

This option produces a list of both generic and customized submodels that are available from the current path list. This list is limited to user submodels only. This means submodels that do not belong to any standard **AMESim** library.

You can select from this list and load into **AMESim** (click on *Load*) or load into **AMECustom** (click on *Customize*) or remove (click on *Remove*) any existing user submodel.

vailable User Submodels				?
Name	Туре	Icon		Load
È-C:/AMETest/submodels	Storage directory		_	
⊡- pmover01	Component icon			Customize
PM60	User submodel			
⊕. elasticendstop	Component icon	±±		
		포	-	<u>R</u> emove
				<u>C</u> lose



4.3.3 Options menu

Figure 4.81: Options menu

<u>O</u> ptions	<u>P</u> reviews	$\underline{T}ools$	<u>W</u> ine
<u>P</u> atł	n list		
<u>C</u> olo	or preferenc	es	
<u>A</u> ME	ECustom Pr	eferenc	es

Path list

Select this item to change the **AMESim** path list. This functionality is precisely the same as in **AMESim**.

Figure 4.82: Path list dialog box

🔊 Path List	? 🗙
The path is used for finding AME.make, AMEIcons, so You can either pick the libraries on the right or add a of the text field and browse button below. Path list should contain at least '\$AME' path.	ubmodels.index. directory using Available categories Standard AMESim library Hydraulic (\$AME/libhydr) Hydraulic Component Design (\$AME/li Hydraulic Resistance (\$AME/libhr) Add to path list
Add user defined path	
Add	
	<u> </u>

Color preferences

Select this item to change colors assigned to categories and lines as well as line styles. This functionality is precisely the same as in **AMESIM**.

Figure 4.83: Color preferences dialog box

Color Preferences			? ×
Categories			
Description:		Color:	
Mechanical	•		<u>D</u> efault
Lines			
Туре:	Style:	Color:	
Signal 👤			D <u>e</u> fault
		<u>2</u> K	<u>C</u> ancel

AMECustom Preferences

Select this item to produce the *AMECustom Preferences* dialog box. Figure 4.84 shows the options set to their default values.

MECustom Preferences
✓ Preview in file browsers Application font Number of files in 'Last Opened Files' list: 4 HTML & equation editor:
Check Submodels Create backup after check: ⓒ ask me ○ always ○ never
Background C [mage:
<u>R</u> estore standard <u>Apply</u>

Figure 4.84: AMECustom Preferences dialog box

The *Preview in file browsers* check box applies to the *Background* area when the *Image* button is selected and the browse button is clicked.

If you click on *Application font*, you can select a different default font for **AMECustom**.

Number of files in 'Last Opened Files' list allows you to change the number of files displayed in the *Files* menu.

You can select the HTML editor you wish to use from **AMECustom** in order to complete or modify the documentation associated with your customized objects.

After a check submodels is applied to a customized supercomponent you have the possibility to create a backup of its original version. You can choose this backup is created always, never or on demand.

The two radio buttons under the label *Background* allow you to change the default main window when no objects are loaded. If *Color* is checked, you can select a particular color. If *Image* is checked, you can enter the path and name of a file to specify an image to be used. Alternatively you can browse for the file and if the *Preview in file browsers* check box is ticked, you will see a reduced size image of the selected file in the browser.

4.3.4 Previews menu

Figure 4.85: Previews menu



Parameter mode



This option takes the active object in its current state and shows how the *Change Parameters* dialog box will look like for this object in **AMESim**. You can also click on the button in the *Toolbar*.

Use this facility frequently when you are customizing submodels and supercomponents as you can identify mistakes at an early stage.

Run mode



This option takes the active object in its current state and shows how the *Variable List* dialog box will look like for this object in **AMESim**. You can also click on the button in the *Toolbar*.

Use this facility frequently when you are customizing submodels and supercomponents so that you can identify mistakes at an early stage.

4.3.5 Icons menu

Ico <u>n</u> s <u>T</u> ool:	s <u>D</u> ocumentation
<u>A</u> dd cat	egory
<u>R</u> emove	e category
A <u>d</u> d cor	nponent
R <u>e</u> move	e component
lcon <u>d</u> e	signer
<u>C</u> hange	object icon

Figure 4.86: Icons menu

Add category

If you do not want to use a component icon from the available **AMESim** libraries, you must create your own component icons and these must be stored in your own category icon. Category icons are the ones you see at the left side of the **AMECustom** window.

The steps involved in creating a new category icon are the following:

1. Select Icons ► Add category...

A browser appears.

2. Select a directory for your category.

Figure 4.87: Browsing for a Folder for an AMESim category.

Browse for Folder	? ×
Select a directory for the new category	
 →	•
OK	Cancel

3. Click on OK.

If the selected directory is not in the **AMECustom** path list, the following dialog box is produced:

Figure 4.88: Adjusting the Path List.



You can then update your path list and you will be asked for the category name and description. As soon as the description is validated, the *Icon Designer* appears. You can now create an icon for the new category.

Remove category

To be able to remove a category

- · there must be no component icons in this category and
- you must have write permission for the appropriate *.xbm*, *.ico* and *AMEIcons* files.
- 1. Select Icons ► Remove category...

The Remove Category dialog box appears:

Figure 4.89: Removing a Category.

🔊 Remove Category	? ×
Select the category you want to remove:	
Mechanical	•
Remove icons files (*.ico, *.xbm)	
	<u>C</u> ancel

- 2. Select the category you want to remove.
- 3. Check the box *Remove icons files* if you want to remove also the files of the icon (*.ico*, *.xbm*).
- 4. Click on OK.

Add component

Step 1: Create a new component

1. Select the Add component item of the Icons menu.

The dialog box shown below is displayed:

Figure 4.90: Adding a New Component Icon.

🛃 Design C	omponent Icon	? ×
		<u>D</u> raw icon
Icon:	No icon specified	Load bitmap
		<u>S</u> elect icon
Name:	<u></u>	
Description:		
Parent category:	Hydraulic, pneumatic and electrical I	ines 💌
		<u>C</u> ancel

2. Fill in the *Name*, *Description* and *Parent category* fields to give the necessary information on your new icon.

Step 2: Specify an icon for the new component.

There are 3 different ways of assigning an icon:

• **Draw a new icon:** Click on the *Draw icon* button to produce the *Icon Designer*. Design the icon and set its ports: please refer to the **AMESim** manual section 6.5.2. Creating a supercomponent icon (step 1 to step 6) to learn how to create an icon and add ports to it. Finally click on the *Save icon to AMESim files* button of the *Icon Designer* dialog box.

- Load a bitmap from a file: A file browser allows you to load a bitmap from a file.
- Select an existing icon from another category: Select the icon you want within a list of existing icons.

When you have designed the icon and set its fields properly, click on the *OK* button.

Remove component

To be able to remove a component icon:

- there must be no submodel or supercomponent associated with the icon and
- you must have write permission for the associated .ico file.

1. Select Icons ▶ Remove component...

The Remove Icon dialog box appears.

Figure 4.1: Removing a Component Icons.

Remove Ic	on	? ×
Select the compo	nent icon you want	to remove:
lcon:		<u>S</u> elect
Parent category:	Mechanical	T
	<u>0</u> K	<u>C</u> ancel

- 2. Select the component icon you want to remove using the Select button.
- 3. Select the parent category of the component icon.
- 4. Click on OK.

Icon designer

The Icon designer facility can be started as a general facility if you select **Icons Icon designer...** It can also be started when creating a new category icon or when creating an icon for a customized object.

With the **AMECustom** *Icon Designer*, you can create and save one or more icons and specify their ports. Please refer to the **AMESim** manual section 6.5.2.Creating a supercomponent icon (step 1 to step 6) to learn how to achieve this.

Change object icon

Use **Icons** ► **Change object icon...** if you want to replace the existing icon which the current active customized object is associated with. You can do the same thing by clicking on the *Change icon* button. In either case an *Icon selection* dialog box is created.



Figure 4.2: The Icon selection dialog box

Lon Selection		? ×
Name	lea	on
- Mechanical	-d ^W) }-
rack2	a a	Parte
cam2	Ę)r-
···· arm	⊳ €	se l
simple_crank	ļ	Å=
	<u>0</u> K	<u>C</u> ancel

Expand the tree structure and select the new icon.

Note that you will be presented with subset of icons. These are the ones that are compatible with the active customized object. It is possible that no icons are suitable! In this case you will have to create one. Naturally you must create your new icon before you can attach a customized object to it.

4.3.6 Tools menu

Tools	Documentation	n <u>W</u> indov
AM	E <u>L</u> exicon	Ctrl+L
Che	eck <u>s</u> ubmodels	
Exp	pression Editor	
Sta	rt AMESjm	
Sta	rt AMES <u>e</u> t	
Lice	ense viewer	

The items on this menu are a reduced set of those available in **AMESim**. The *Check submodels* item needs further explanation in the special context of **AMECustom**.

AMELexicon

You can use *AMELexicon* at any time while working with **AMECustom**. The main function of the *AMELexicon* utility is to help you to make the titles of the parameters and variables consistent. When you create your own customized objects, it is extremely easy to have titles that almost the same as the titles in other related generic or customized objects. Unfortunately 'almost' is not good enough for facilities like *Common Parameters*!

Basic rules

- When you select a library in the path list within the *Build* tab, *AMELexicon* scans the submodels of this library.
- You have the option to restrict the search by filtering according to submodel name, variable/parameter title and unit.
- Results of a search are shown under the *Titles* tab.
- In the *Titles* tab you have a *Variable titles* and a *Submodels list*. The *Submodels list* displays the submodels with titles with in the *Variable titles*.
- *AMELexicon* recognizes the difference between variables/parameters with identical titles but different units
 - You can drag and drop the title you want onto a variable/parameter in you active AMECustom object.
- If you double-click on a submodel in the *Submodel lists*, **AMECustom** attempts to open **AMESet** with the submodel loaded.
- If a title seems to appear twice in the list with the same unit, this probably means that there is a spelling mistake on one of the two titles.

AMELexicon dialog box

To produce AMELexicon dialog box, select the menu Tools AMELexicon...

When you open the AMELexicon dialog box, the Build tab is displayed first.

AMELexicon		
Titles Build		
Path List:		
Path		
SAME		
L h:/amesim/submode	els	
Colori Norra		CalaakAll
Select None	Update Path List	Select All
Submodel Filter:		
×		
C Variables	C Parameters	Bot
- Title filter		
🗖 Case Sensitive	📃 Exact Mat	ch
Linit filter		
	Generate	

Figure 4.91: Build tab

The Build tab contains four groups of options:

Path list

This is the current **AMECustom** path list.

The buttons Select None and Select All are obvious.

The button *Update Path List* is useful if you have modified your path list and you want to update it in this dialog box.

Submodel filter

In the white field you can enter a name of a submodel from which you want to check the parameter and variable titles. Alternatively you can use wildcards like * and ? to specify a collection of submodels.:

To find	Do this	Example
Zero or more characters, excluding spaces and punctuation	Enter an asterisk (*).	HL*4 finds submodels such as HL04 and feet.

To find	Do this	Example
Any single character, excluding spaces and punctuation	Enter a question mark (?).	F??4 finds submodels with four-letter names that begin with F and end with 4for example, FP04.

You can filter variables, parameters or both.

Title filter

You can enter one or several words in the white field.

If you tick the *Case sensitive* box, *AMELexicon* will take into account the case of the words you typed in.

If you tick the *Exact Match* box, *AMELexicon* will take into account the exact text string you typed otherwise it takes the words in the string as key words.

Unit filter

You can type several units in the white field separated by a space. *AMELexicon* will give you the list of parameters and variables which have the same units.

Generate button

When you have entered the characteristics of the search, click on the *Generate* button. This produces the *Titles* tab containing the title list corresponding to your search. By selecting a particular title, the *Submodel list* is reconstructed to show the submodels associated with this title.

AMEL	exicon				? >
Titles	Build				
Variab	e titles:				
Ref	Title	Unit			
1	final speed	rev/min			
1	prime mover speed	rev/min			
1	startup time	s			
1	time constant	\$			
Submo	idel list:				
Icon	Library			Submodel	Place
meca	h:/amesim/submodels/submodels		dels	TEST_PM50 real par	
•					•
Print	Print Title List			Print Submodel List	
					<u>C</u> lose

Figure 4.92: Titles tab

You can drag and drop a title from *AMELexicon* to a title of a parameter or a variable within **AMESet** or **AMECustom**. This can be done only for titles. If

you place the cursor in the wrong place, it becomes a $\boldsymbol{\heartsuit}$.

Check submodels

In **AMESim** *Check Submodels* is used to check the active **system** or **generic supercomponent**. The idea is to provide a mechanism for painlessly updating the system or supercomponent with respect to changes in the specification of the constituents.

In **AMECustom** the check can be applied to the active:

- customized submodel or
- customized supercomponent

In the first case, the specification file of the customized submodel is compared with the specification file of its generic version. This is what we call a *Check consistency*.

In the second case a full *Check submodels* is applied as in **AMESim**, since a supercomponent is just like a piece of an **AMESim** system.

Basic rules

The following basic rules apply to *Check Submodels* in **AMECustom**:

- Check Submodels starts automatically when you load an object.
- When it starts automatically and there is no problem, no dialog box appears.
- You can also start it manually using **Tools** ► **Check Submodels...** you should do this when you are altering a generic object or its constituents and simultaneously its customized version is used in **AMECustom**.
- If you start *Check Submodels* manually and have made changes to the object in **AMECustom**, do a *Save* first.

Check submodels of a customized submodel

This is the simplest case, **AMECustom** compares the specifications of the customized submodel with the specifications of its generic version.

- If there are important differences (the number of ports has changed) you should completely rebuild the customized submodel.
- If there are just differences between the titles of the variables or parameters, **AMECustom** will ask you to make the link between the customized and the generic versions:


Figure 4.93: Check customized submodel

• In other cases, **AMECustom** will give you the possibility to update the customized version according to the new specifications of the generic submodel. This can happen when the unit of a parameter or variable has changed, or when the number of parameters or variables is different:

Figure 4.94: You can update the customized version



Check submodels of a customized supercomponent

It you start the process manually, a Check Submodels dialog box is produced:

Figure 4.95: Check submodels dialog box

B-Check 'CUSVALVES50' Submodels		? ×
Circuit version: 4.2.0 AMESim version: 4.2.0		<u>D</u> etails >>
Checking:		Start
	0%	<u>C</u> lose

You can click on *Start* with the dialog box in this state or expand it by clicking on *Details*: the dialog box then takes on the form shown below.

∰ Check 'CUSVALVES50' S	ubmodels			? ×
Circuit version: 4.2.0 AMESim version: 4.2.0 Checking:			03	<< <u>D</u> etails Start Close
Submodels <u>R</u> eport				1
Submodel	Check	Update	Туре	Access Path
庄 🔂 Circuit			circuit	
Use this path for <u>o</u> ther sub	models			
☑ Try to use same path as in cj	rcuit			
lgnore Update	Update	all		

Figure 4.96: Check submodels extanded dialog box

Note the *Try to use same path as in circuit* check box. By default this is enabled. Normally we expect supercomponent constituents not to have moved and so you do not need to change the default. However, if you move a supercomponent from one computer system to another, the paths used in the supercomponent file is likely to become irrelevant and hence you should disable this check box.

Note also the tabs labeled *Submodels* (selected by default) and *Report*. The first show the tree structure of the supercomponent circuit as the check proceeds and the second is more detailed and contains no graphics. It is of a form that can be printed.

Submodel location

AMECustom checks the constituents of the loaded supercomponent against their specification. This involves finding the constituent *.spe* files (and *.sub* files if appropriate). It locates these from the specification of the loaded supercomponent, from the current **AMECustom** path list or, if it is unsuccessful, it will ask for your help in locating these files.

- **Note:** It is worth reviewing the current path list first and adjusting it if necessary.
 - In most situations the *Try to use same path as in circuit* check box should be left checked. However, if the object loaded into AME-Custom has just been moved from one computer system to another, its path name may be wrong and it is more efficient if this box is unchecked.
 - If **AMECustom** needs help finding a file, it will tell you the name of the file. You can then enter the path in the input box or use the *Browse* button to search for it.
 - If you think that this path may need to be used again, check the *Use this path for other submodels* box.

Check submodels starting

Start the process by clicking on *Start*. A check finished message is displayed with no report of any problem if the supercomponent is up-to-date.

However, the process will stop if a problem is found. There is an entry in the *Report* window explaining the nature of the problem. Note that the procedure is the same as for an **AMESim** model. For more details please refer to the *Check submodels* section of the **AMESim** manual.



It is vitally important that after an update you examine the customized object very thoroughly.

Remember that it is not possible to reconstruct an egg from an omelette! For catastrophic changes it may be better to remove the old customized object and create a new one.

Expression Editor

Start of the Expression Editor

There are two different ways to start the Expression Editor:

- when setting a value for a parameter or variable, click on _____ to display the *Expression Editor*,
- use the menu Tools ► Expression Editor...



Figure 4.97: Expression Editor

Use of the Expression Editor

You can use the Expression Editor for:

- calculation,
- entering mathematical functions and expressions as a parameter value.

Procedure

- 1. To enter a mathematical function in the input box, select a function in the *Mathematical functions* list.
- 2. Double-click on the mathematical function.

The mathematical function is placed in the input box.

3. Enter one or several values in the brackets.

You can select values in the *Fundamental constants* and *Global parameters* lists by double-clicking on the values.

- 4. If necessary complete the expression by selecting operators and functions in the two lists.
- 5. When your expression is complete, click on the *equal* button.

The result is displayed, when possible, in the grey box next to the *equal* button.

Expressions

The expressions that you can enter in the input box can be made up of:

- global parameters;
- real and integer constants;
- the label PI which is taken to be an approximation to π ;
- the arithmetic operations +, -, *, / and for raising to a power ^ or **;
- the boolean operations: !, !=, &&, ||, >, <, >=, <=,==;
- parentheses '(' and ')' with their usual mathematical significance;
- coma ',' for seperating variables;
- the following functions of one variable:

sin	COS	tan	acos	asin	atan	log	log10
sinh	cosh	tanh	acosh	asinh	atanh	exp	abs
sqrt	integ	differ	lsqrta	fabs			

• the following functions of two variables:

atan2 sign

• and the following functions of two or more variables:

min max

AMESim will first check your expression. If it finds it is acceptable, it will enter the expression as a value of a parameter if you started the *Expression Editor* from the value field of a parameter or variable.

Note: The maximum size for an expression is 255 characters.

Start AMESim/Start AMESet

Just click on these items to start **AMESim** or **AMESet**.

License viewer

Select this item to see who is currently using **AMESim** product licenses. Normally this is only useful for clients who have multiple licenses. Note that you do not use a license for an **AMESim** library to use this library in **AMECustom**. For example you can use a pneumatic library icon and make a copy of a pneumatic library submodel for customizing without using a pneumatic library license token.

License Viewer					? X
Feature used	Index	Max. no.	User	Machine	
AMECustom	1	100	Support	refw2k	
AMESet	1	100	CML	image10	
AMESet	2	100	CRoman	desketudes3	
AMESet	3	100	EDoming	image9	
AMESet	4	100	ARoussel	aroussel	
AMESim	2	100	Support	refw2k	-
,					

Figure 4.98: License Viewer displays the available features

4.3.7 Documentation menu

Figure 4.99: Documentation menu



Create HTML skeleton

Select this item to create HTML documentation for the active customized object.

- **Note:** The name of the file is inherited from the customized object with an *html* extension.
 - This .*html* file is put in the *doc* directory of the **AMESim** node in which the customized object is stored.
 - The .*html* file is constructed from the specification (.*spe*) and source code (.*c* and *f*, if any) of the generic version of the customized object. For customized submodels, fill in the full description part of the source code as this forms a major part of the documentation.

Edit HTML documentation

Use **Documentation** Edit HTML documentation to modify and complete the various sections of your customized object documentation. If no *HTML* editor is specified in the **AMECustom** preferences, you will be asked to select one.

View HTML documentation

Use **Documentation** ► **View HTML documentation** to view the documentation of the active customized object. In contrast if you use **Help** ► **OnLine**, you get documentation on all submodels.

4.3.8 Windows menu

Use the Windows menu to modify the display of your systems.

Figure 4.100: Windows menu

<u>W</u> indows <u>H</u> elp
🔁 <u>C</u> ascade
d:/ametest/tutorial/submodels/INCAM50
Close <u>a</u> ll

Cascade

This option arranges the windows corresponding to the customized objects currently used so that they overlap with their titles remaining visible.

List of currently opened files

This part of the Windows menu shows the list of currently opened files.

You can bring an object to the top by selecting it, it becomes the active object.

close all

This option initiates the process of closing all the windows associated with customized objects. If there are unsaved changes, you will be asked to save them.

4.3.9 Help menu



Figure 4.101: Help menu

Online

Select this item to view the documentation of **AMESim** submodels. These may be in libraries supplied as an **AMESim** product, generic or customized submodels or supercomponents produced by you or by a colleague. At the left side of the screen you can find the three following tabs:

- Contents, which shows the list of the documentation topics.
- Index, from which you can type in a keyword and get the related entries.

Search, from which you can get the list of all the documents contain-٠ ing at least one occurence of a given keyword.



Figure 4.102: On-line help window

About

This option gives a lot of information on the version of AMECustom you are using and the libraries to which you have access. If you contact the AMESim hotline, you may need to give information from this dialog box to the support team.

The AMECustom Toolbar 4.4

Each button of the toolbar is equivalent to a menu item of the menu bar.



File > Open or Ctrl+O (see Open)







Previews > Run mode (see Previews menu)

Ж Edit Cut or Ctrl+X (see Edit menu)

Edit Copy or Ctrl+C (see Edit menu)

Edit > Paste (see Edit menu)



4.5 The Category buttons

The category buttons are located at the left side of The AMECustom main window. When you click on one of these, a dialog box of component icons appears (Figure 4.103) which will be familiar with all users of **AMESim**, **AMERUN** and **AMESet**.





In **AMECustom**, when you select one of these icons you get a *Submodel List* dialog box (Figure 4.104).

🍇 Submodel List		? ×
2 -21- 1	∶am2	
Submodel list	Description	Туре
CAM00 EXCAM50 GENSUPERCA CUSTOMCAM5	cam and cam follower cam and cam follower M50 generic supercomponent 1 generic supercomponent	generic submodel customized submodel cam generic supercomponent cam customized supercomponent
E <u>x</u> ternal variables	<u>H</u> elp	Explore
		<u>K</u> ancel

Figure 4.104: Submodel List dialog box

This contains a list of generic and customized submodels and supercomponents associated with the selected icon that **AMECustom** can find using the current path list. You can select one item in this list.

• Click on *OK* to load the object into **AMECustom**. If it is encrypted, you will be asked for a password.

Figure 4.105: You must know the password if the submodel is encrypted

A Enter Password		? ×
Submodel is protected. Please enter the password to be able to use it	t.	
	OK	Cancel

- The *Explore* button is sensitive only if it is a supercomponent. Click on this to see the supercomponent constituents. If it is encrypted, you will be asked for the password.
- Click on *External variables* to display the external variables of the object.
- If you click on *Help*, **AMECustom** will attempt to load and display documentation on the selected object.
- Click on *Cancel* if you have lost interest in the objects displayed.

4.6 The customized object general features

4.6.1 General features of a submodel

These are shown on the left hand side of the **AMECustom** main window when a **submodel** is selected for customization:



Figure 4.106: General features of the submodel

Some Right-click menus (page 90) are also available.

Name

When a new customized submodel is being created, **AMECustom** assigns a default name to it. If the name of the associated generic submodel is *GENNAME*, then the default name is *UNNAMED_GENNAME*. The first time you save your customized submodel, **AMECustom** asks you to replace the default name by a new one. Submodel name conventions are discussed in the introduction of Chapter 2:Customizing submodels (page 7).

lcon

The default icon of a customized submodel is the same as the icon of the generic submodel it is associated with. However you can change this using the *Change icon* button:





This produces an *Icon selection* dialog box which allows you to select a different icon. Note that you will be presented with subset of icons. These are compatible with the active customized submodel. It is possible that no icons are suitable! In this case you will have to create your own one. Naturally you must create a new icon before attaching any customized submodel to it.

Brief description

This appears in an editable field which contains, by default, the generic submodel description. You should replace this by the description of your customized submodel.

Full description

If you click on the button labeled *Full description*, the dialog box below appears. By default this shows the full description of the associated generic submodel. You must alter it in order to take into account the modifications done in your customized submodel.

Ŷ	& Full Description	? ×
	IITLE : EXCAM50 DESCRIPTION : This is a submodel of an exhaust cam profile with cam follower. An ASCII file is read defining the linear displacement of the contact point (in mm) according to the angular displacement (in degrees, in the range 0 to 360). Note that in this file, the displacement at the first angle value must be equal to the displacement at the last angle value. On port 1 the outputs are the linear speed in m/s and the displacement in m, the input is a force in N. On port 2 the output is a torque in Nm and the input is a speed in rev/min.	
	Clear all Generic description Load default From file <u>D</u> K <u>Cancel</u>	

This dialog box also contains the four buttons described below:

- *Clear all*: delete the content of the description.
- *Generic description*: replace the text currently shown by the description of the associated generic submodel.
- *Load default:* replace the text currently shown by a template containing the headers TITLE, DESCRIPTION...
- *From file*: produce a file browser from which you can pick up a text file containing a description. This could be a description of the specific customized object you have produced in another editor or your own private template that you prefer to use.

4.6.2 General features of a supercomponent

These are shown on the left hand side of the **AMECustom** main window when a **supercomponent** is selected for customization:

CUSVALVES50						
twinvalves						
Change icon	Ch <u>a</u> nge icon					
Brief description:						
twin valves in a cylinder	head					
Eull description						
Supercomponent						
Submodel Type						
E-CUSVALVES50 supercomponent						
•• EXCAM50	customized object					
0 F000	component					
0 F000	component					
@INCAM50	customized object					

Figure 4.109: General features of the supercomponent

Some Right-click menus (page 90) are also available.

Name, Icon, Brief description, Full description

These features are identical to the ones described in previous section.

List of constituents

This list contains two columns indicating the icon, the submodel name and the type of each constituent (component, line or customized object) of the selected supercomponent.

Note that when you select a constituent in this list, the table of the *Internal items* is updated accordingly (see section The Internal items lists (page 92)).

Below this list the *Explore* button produces a dialog box showing the circuit associated with the supercomponent. If you click on a constituent of this circuit then the corresponding line is selected in the list and the table of the *Internal items* is updated accordingly. **AMECustom does not allow you to change the circuit of a supercomponent**.

E	Explore Superco	mponent	? ×
		CUSVALVES50-1 twin valves in a cylinder head	
	Inlet		
	↓	》 《	_
	E <u>x</u> ternal variables		<u>Print</u>

Figure 4.110: Circuit associated with the supercomponent

The *Explore Supercomponent* dialog box also contains the following two buttons:

• External variables: show the external variables of the supercomponent.

Figure 4.111: External variables of the supercomponent

🗞 External Variables	? ×
CUSVALVES50	
twin valves in a cylinder head	
For variables which have a direction associated with them positive sign is in the direction of the arrow.	12
1 → Nm ← rev/min	

• *Print*: print the supercomponent circuit.

4.6.3 Right-Click Menus

- Main right-click menus
- Main functions available from the right-click menus

Main right-click menus

Different right-click menus are available from the main interface depending on the part you click on.

AMECustom - [H:/MAS005]	
S File Edit Options Previews Icops Icols Windows Help	
UNNAMED_MASO05 Image: con Image: con Brief description: Brief descr	

Figure 4.112: the different right-click menus

	Right-click on	to access the following commands
1	the left part of the display	• hide all
		• show all
	the brief description	• undo, redo last action
2		• cut, copy, paste, clear text
		• select all text

3	a submodel global parameter	 cut, copy, paste real global parameter remove, add new real global parameter move up, move down real global parameter
		• hide all, show all real global parameter
4	an external item title or value	• reset default value, reset initial value
		• undo, redo
		• cut, copy, paste, clear
		• select all
5	an internal item	 copy internal variable, paste internal or external variable
		• hide, show all internal variables

Main functions available from the right-click menus

Modifying the brief description

If you right-click on the object brief description, a menu appears. This is similar to the menu produced with most text (as opposed to numerical) fields and it is shown in Figure 4.113. It contains items the function of which is obvious. These menu items apply to the text in the *Brief description* field.

Figure 4.113: Right-click menu

<u>U</u> ndo	Ctrl+Z
<u>R</u> edo	Ctrl+Y
Cu <u>t</u>	Ctrl+X
<u>С</u> ору	Ctrl+C
<u>P</u> aste	Ctrl+V
Clear	
Select All	Ctrl+A

Hiding status of variables and parameters

We can alter the status of an individual internal variable or parameter using the special check boxes but often it is useful to operate on a whole group of variables and parameters. We can do this using various right-click menus.

Figure 4.114: Hidden status check boxes

Hidden
N
V

Changing the hidden status of all variables and parameters of a customized submodel

When the object being customized is a submodel, the area below the *Full description* button is blank but a right-click produces a menu. This can be used to hide or show all parameters and internal variables of the current object.

Figure 4.115: Right-click menu in the Full description area

Brief description:					
inlet cam and cam follower					
Eull description					
	<u>H</u> ide all				
	<u>S</u> how all				

The same menu can be produced with a right-click in the internal item area.

Figure 4.116: A right-click menu is available in the internal item area

Internal items:						
l <u>n</u> ternal variables	R <u>e</u> al p	arameters	Inte	ger par	amete	
Title				Unit	Тура	
angular displacement of the cam degree exp						
angular displacement	t 0 🕞	Copy intern	al var	iable		
angular displacement	to 😐	<u>C</u> opy intern		Iable		
transformer ratio	B	<u>P</u> aste			•	
		<u>H</u> ide all inte	ernal v	/ariable	s	
		<u>S</u> how all in	ternal	variable	es	

Changing the hidden status of all variables and parameters of a customized supercomponent

When the object being customized is a supercomponent, right-button menus are available using:

- the supercomponent tree structure,
- the Explore Supercomponent dialog box or
- the internal item area.

For the supercomponent tree structure, select (highlight) an item in the tree structure and operate the right button. If the selected item is the supercomponent being customized (i.e. the root of the tree), the hide/show applies to **all** the constituents. If the selected item is a constituent, it applies only to this constituent.

E	Supercomponent-			
Submodel		Туре		
	⊡-terCUSVALV		All constituents	
		Copy parameters		
		🖹 Paste parameters	Supercomponent	
			Submodel Type	
			È- ₩ CUSVALVES50 supero	comp
			EXLAMOU Custor F000 comp	mizea onent
	Selected constituent only		ly	
				s
				rs:

Figure 4.117: Show or hide the constituents

For the *Explore Supercomponent* dialog box, if a constituent component is selected as in Figure 4.118, a right-click menu applies only to the selected component. If no component is selected, it applies to all constituent components.

Figure 4.118: Right-click menu applied to a selected component

	Component variables and parameters	•	Hide all
	External variables		Show all
Ó	Labels	۲	Copy parameters
	Bird's eye view		Paste parameters
	Help		

You can also hide or show labels with this menu, get help on a constituent component or get a bird's eye view (for big supercomponents).

The right button menu for the *Internal items* area is active only if a constituent component is selected and the result applies only to this component.

4.7 The global parameter lists

These lists show the global parameters currently defined for the selected customized object (submodel or supercomponent). It contains the following three tabs: *Real, Integer, Text.*

	Supercomponent global parameters:								
ĺ	Real Integer Iext								
	Name	Title	Unit	Minimum	Default	Maximum	Hidden		
VPRELOAD valve spring preload m 200 300 800 🔽									
VSTIFF valve spring stiffness N/m 10				10000	100000	1e+006			
VCLEAR valve clearance n			m	5e-005	0.0001	0.0002			
l	VMASS valvemass kg 0.005 0.01 0.05						<u>A</u> aa new		
	VMAXLIFT	maxi possible valve	m	0.02	0.03	0.085		Re <u>m</u> ove	

Figure 4.119: Supercomponent global parameters

You can:

- Add a global parameter by clicking on the *Add new* button after selecting the tab associated with its type. When it is created, default values are assigned to each of its characteristics.
- **Delete** a global parameter by clicking on the *Remove* button after selecting it in the list.
- Move up and down a global parameter using the arrow buttons. However, when you save **AMECustom** does a check to ensure a variable is not expressed in terms of another later in the list.

You cannot:

• Express a global parameter in terms of itself.

Real

Real global parameters have the characteristics described below:

Name

Click on the column *Name* and type in a name for the real global parameter. This name will be used in the column *Default value* of the *External variables* or *Internal items* list.

Title

Click on the column *Title* and type in a title for the real global parameter. This title is a short description of the parameter which will appear in **AMESIM**.

Unit

Click on the Unit column and type in the unit of the real global parameter.

Alternatively you can click on the _____ button in this column, and you will get a dialog box for selecting the unit.

Domain	U	nit 🔄
nagnetomotive voltage		
🛱 - mass		
	g	
	kg	
	Ь	
	m	9
-	M	g _
·	to	nne
⊕-moment of inertia		_
i∓irotarv acceleration		
kg		<u>F</u> ind

Figure 4.120: Unit chooser dialog box

Minimum value, Maximum value and Default value

These characteristics are assigned by entering a value in the corresponding column. If the three values are identical, the real global parameter will not appear in **AMESim** and it will not be possible to change its value.

Hidden

Put a tick mark in this check box if you do not want this global parameter to be available from the **AMESim** *Change parameters* dialog box.

Integer

Integer global parameters have the characteristics described below:

Туре

Select *Standard* or *Enumeration*. An enumeration parameter can have several values and each value is associated with a text string. This association is done in the *Enumeration list* column.

Name

Click on the column *Name* and type in a name for the integer global parameter. This name will be used in the *Default value* column of the *External variables* or *Internal items* list.

Title

Click on the *Title* column and type in a title for the integer global parameter. This title is a short description of the parameter which will appear in **AMESim**.

Enumeration list

This column is editable only if the type is set to *Enumeration*. Click on the ______ button in this column, and you will get a dialog box for defining and associating each value of the parameter to a text string.

Figure 4.121: Configure Enumeration dialog box

Ņ	😽 Configure Enumeration						
E	numeral	tion list fo	r 'mode	for data o	ut of range':		
	Value	Text					
1		extrapola	ation				
	2	extreme *	value				
3	3	cyclic					
1							
	Ad	d	<u>B</u> e	emove	1	↓	
				<u>0</u> K	<u>C</u> an	cel	

Minimum value, Maximum value and Default value

These characteristics are assigned by entering a value in the corresponding column for a standard parameter. If the three values are identical, the standard integer global parameter will not appear in **AMESim** and it will not be possible to change its value.

For enumeration integer global parameters, only the default value is required.

Hidden

Put a tick mark in this check box if you do not want this global parameter to be available from the **AMESim** *Change parameters* dialog box.

Text

Text global parameters have the characteristics described below:

Name

Click on the *Name* column and type in a name for the text global parameter. This name will be used in the *Default value* column of the *Internal items* list (in the tab labeled *Text parameters*).

Title

Click on the *Title* column and type in a title for the text global parameter. This title is a short description of the parameter which will appear in **AMESIM**.

Default value

Click on this column and type in the text you want to assign. Often text parameters are used to specify the full pathname of a file. In this case, you can

use the button to browse the default file instead of typing its name and path:

Figure 4.122: Browser of the Open dialog box

Open					?	×
Look jn:	🔄 Custom	•	£	d *		
📄 submod	els					
🗟 camder	no2.ame					
🛛 🖻 camden	no2_					
🛯 🔊 submod	els.index					
1				_		
File <u>n</u> ame:					<u>O</u> pen	
Files of tune:	All files (*)		-		Connel	1
, 100 of <u>3</u> 7po.					Lancei	┙╽

Hidden

Put a tick mark in this check box if you do not want this global parameter to be available from the **AMESim** *Change parameters* dialog box.

Using global parameters

Global parameters of appropriate type can be assigned as default values to:

- Any internal variable of the customized object, provided it has a default value. This means it must be an explicit or implicit state, a fixed variable or a constraint which is accessed in the *Internal items* area. The global parameter must be of integer or real type. The original internal variable will be hidden and the global parameter will appear in the *Change Parameters* list.
- Any real, integer or text parameter of the customized object. In this case, a tick mark is automatically added in the corresponding *Hidden* column and this tick mark cannot be removed. This means that this item will not appear in **AMESim** anymore, it will be replaced by the global parameter.

h	nternal items:								
	I <u>n</u> ternal variables	R <u>e</u> al parameter	s	Iņ	teger param	neters	Te <u>x</u> t p	parameters	
	Title		Uni	t	Minimum	Default		Maximum	Hidden
I	CRANK0-1: radius of	crank	mm		0.1	R/	ADIUS	10000	$\overline{\mathbf{A}}$
l	CRANK0-1: length of	connecting rod	mm		0.2	LE	NGTH	15000	
l	CRANK0-1: offset for	displacement	mm		-1e+006	0	FFSET	1e+006	
l									
Į									

Figure 4.123: Real parameters tab

• Any external variable of the customized object, provided it can be assigned a default value. However, since external variables cannot be hidden and both the global parameter and the external variable in its own right will appear in the *Change Parameters* list. It is normally better to customize an external variable in its own right rather than using a global parameter.

Figure 4.124: External variables

l	Extern	ial variables:					
	Port	Title	Unit	Туре	Minimum	Default	Maximum
l	1	velocity at port 1	m/s	explicit state variable	-1e+006	0	1e+006
l	1	displacement port 1	m	explicit state variable	-1e+006	DISP0	1e+006
l	1	acceleration at port 1	m/s/s	variable			
	1	force at port 1	N	variable			
	2	force at port 2	N	variable			

Right-click menus

Title / Value

If you select the title or a value (Minimum, Default or Maximum) of a global parameter and you right-click on it, a menu similar to the one shown in Figure 4.125 appears.

Figure 4.125: Right-click menu

Reset default Reset initial v	value alue
<u>U</u> ndo	Ctrl+Z
<u>R</u> edo	Ctrl+Y
Cu <u>t</u>	Ctrl+X
<u>С</u> ору	Ctrl+C
<u>P</u> aste	Ctrl+V
Clear	
Select All	Ctrl+A

This menu contains items which functions are obvious. They apply to the selected title or value.

4.8 The External variables list

This list shows the external variables of the customized object (submodel or supercomponent) currently selected.

Extern	al variables:						
Port	Title	Unit	Туре	Minimum	Default	Maximum	
1	velocity at port 1	m/s	explicit state variable	-1e+006	0	1e+006	
1	displacement port 1	m	explicit state variable	-1e+006	0	1e+006	i
1	acceleration at port 1	m/s/s	variable				
1	force at port 1	N	variable				
2	force at port 2	N	variable				

Figure 4.126: External variables list

In this section, you can have information about:

External variable characteristics

Port

This column shows the port number associated with external variables. This characteristic cannot be changed.

Title

This column shows the generic short descriptions of external variables. You are allowed to modify them for customizing the object.

Unit

This column shows the unit associated with external variables. This characteristic cannot be changed.

Туре

This column shows the type of external variables. This characteristic cannot be customized.

Minimum value, Maximum value and Default value

These characteristics are initially set to their generic values but you can modify them for customizing the object. If the three values are identical for a given external variable, it will not be possible to change its value in **AMESim**. The default value can be assigned an integer or real global parameter.

Right-click menus

Title / Value

If you select the title or a value (Minimum, Maximum or Default) of an external variable and you right-click on it, a pulldown menu similar to the one shown in Figure 4.127 appears.

Figure 4.127: Right-click menu

Reset default Reset initial v	value alue
<u>U</u> ndo	Ctrl+Z
<u>R</u> edo	Ctrl+Y
Cu <u>t</u>	Ctrl+X
<u>С</u> ору	Ctrl+C
<u>P</u> aste	Ctrl+V
Clear	
Select All	Ctrl+A

This menu contains items the functions of which are obvious. They apply to the selected title or value.

4.9 The Internal items lists

These lists show internal variables, real parameters, integer parameters and text parameters currently defined for:

- the current customized submodel or
- the selected constituent of the current customized supercomponent.

It contains four tabs: *Internal variables*, *Real parameters*, *Integer parameters* and *Text parameters*.

Figure 4.12	8: Internal	items list
-------------	-------------	------------

Internal ite	ms:										
I <u>n</u> ternal v	variables	R <u>e</u> a	l param	eters	Integer paramet	ers	Te	<u>x</u> t parame	ters		
Title			Unit	Туре		Minim	num	Default	Maxir	num	Hidden
angular d angular d transform	isplacemen isplacemen er ratio	tof tof	degree degree m	explicit multi-lir multi-lir	state variable ne macro variable ne macro variable		0	0		360	

In this section, you can have information about:

Internal variables

Internal variables have the characteristics described below:

Title

This column shows the short descriptions of internal variables. They may include a prefix to make the title unique. You are allowed to modify them for customizing the object.

Unit

This column shows the unit associated with internal variables. This characteristic cannot be changed.

Туре

This column shows the type of external variables. This characteristic cannot be changed.

Minimum value, Maximum value and Default value

These characteristics are initially set to their generic values but you can modify them for customizing the object. If the three values are identical for a given internal variable, it will not be possible to change its value in **AMESim**. The default value can be assigned an integer or real global parameter.

Hidden

Put a tick mark in this check box if you **do not** want the selected internal variable to appear in **AMESim**. This option is automatically selected if you assign a global parameter to the internal variable.

Real parameters

Real parameters have the characteristics described below:

Title

This column shows the short descriptions of real parameters. They may include a prefix to make the title unique. You are allowed to modify them for customizing the object.

Unit

This column shows the unit associated with real parameters. This characteristic cannot be changed.

Minimum value, Maximum value and Default value

These characteristics are initially set to their generic values but you can modify them for customizing the object. If the three values are identical for a given real parameter, it will not be possible to change its value in **AMESIM**. The default

Chapter 4 Reference Guide for AMECustom

value can be assigned an integer or real global parameter.

Hidden

Put a tick mark in this check box if you do not want the selected real parameter to appear in **AMESim**. This option is automatically selected if you assign a global parameter to the real parameter.

Integer parameters

Integer parameters have the characteristics described below:

Title

This column shows the generic short descriptions of integer parameters. You are allowed to modify them for customizing the object.

Minimum value, Maximum value and Default value

These characteristics are set to their generic values but you can modify them for customizing the object. If the three values are identical for a given integer parameter, it will not be possible to change its value in **AMESim**. The default value can be assigned an integer or real parameter.

Hidden

Put a tick mark in this check box if you do not want the selected integer parameter to appear in **AMESim**. This option is automatically selected if you assign a global parameter to the integer parameter.

Text parameters

Text parameters have the characteristics described below:

Title

This column shows the short descriptions of text parameters. They may include a prefix to make the title unique. You are allowed to modify them for customizing the object.

Default value

This characteristic is set to its generic value but you can modify it for customizing the object. You can directly type in its value, or it can be assigned a text global parameter. Often text parameters contain a file name associated

with a path. In this case, you can use the _____ button to browse the default file instead of typing its name and path:

Open					? ×
Look jn:	Custom	•	£	d *	
📄 submod	els			_	
a camden	no2.ame				
submod	102_ Iels index				
	olo.indox				
File <u>n</u> ame:					<u>O</u> pen
Files of type:	All files (*)		-		Cancel
	J				Cancel

Figure 4.129: Browser of the Open dialog box

Hidden

Put a tick mark in this check box if you do not want the selected text parameter to appear in **AMESim**. This option is automatically selected if you assign a global parameter to the text parameter.

Right-click menus

Title / Value

If you select the title or a value (Minimum, Maximum or Default) of an internal item and you right-click on it, a pulldown menu similar to the one shown in Figure 4.130 will appear.

Figure 4.130: Right-click menu

Reset defaul Reset initial v	t value value
<u>U</u> ndo	Ctrl+Z
<u>R</u> edo	Ctrl+Y
Cu <u>t</u>	Ctrl+X
<u>С</u> ору	Ctrl+C
<u>P</u> aste	Ctrl+V
Clear	
Select All	Ctrl+A

This menu contains nine items the functions of which are obvious. They apply to the selected title or value.

Hide/Show

If you right-click on any column title in the tab of an internal item, the following menu appears:

Figure 4.131: Right-click menu in the tab of an internal item

Hide all internal variables Show all internal variables

This menu provides two facilities allowing you to hide or show all internal variables displayed in the tab currently selected.

Index

Symbols		
.state file		 43
.var file		 43
۵		
Activity variables		43
AMECustom	•••	
What is it?		1
AMECustom Preferences		 57
AMELexicon		 63
Path list		 . 64
Submodel filter		 . 64
Title filter		 65
Unit filter		 65
AMELexicon dialog box		 63
AMEPack		 5
AMERun		 1
AMESet		 2
AMESim		 1
D		
B Drief description		70
Sotting	•••	 /8
Setting	•••	 10
С		
Cascade		 73
Category buttons		 75
Change Parameters with customized submodels		 14
Check submodels		 66
starting		 69
Close all		 73
Color preferences		 56
Create a new component		 60
Customized objects		 2
Distributing to other users		 5
Important rules		 5
List of		 54
Removing		 54
Customized submodel		
Example in thermal hydraulic library		 2
Customized submodels		
Special files		 15
Customized supercomponent		
Cannot explore contents in AMESim		 4
Customizing Supercomponents		 28

Edit menu	
Add new	2
Available customized	4
Available supercomponents 5	4
Available user submodels	5
Сору	2
Cut	2
Delete	2
Hide all	3
Move down	3
Move up	3
Paste	2
Show all	3
Update categories	3
Encrypted customized supercomponent	5
Encryption of customized objects	4
Explore Supercomponent	0
Explore Supercomponent dialog box	9
Expression editor	9
Valid expressions	1
External variables	
Modifying	1
External variables list	1
F	
r File menu	
riie menu	
Class	1
Close	1
Close	1 9
Close 5 Open 4 Quit 5	1 9 1
Close 5 Open 4 Quit 5 Save 5	1 9 1 0
Close 5 Open 4 Quit 5 Save 5 Save as 5	1 9 1 0
Close5Open4Quit5Save5Save as5Flat systems2Flat systems2	1 9 1 0 4
Close5Open4Quit5Save5Save as5Flat systems2Full description2	1 9 1 0 4
Close5Open4Quit5Save5Save as5Flat systems2Full description1Setting1	1 9 1 0 0 4 0
Close5Open4Quit5Save5Save as5Flat systems2Full description2Setting1Full description for customized object7	1 9 1 0 0 4 0 8
Close5Open4Quit5Save5Save as5Flat systems2Full description2Setting1Full description for customized object7G	1 9 1 0 0 4 0 8
Close5Open4Quit5Save5Save as5Flat systems2Full description2Setting1Full description for customized object7GGeneral features	1 9 1 0 0 4 0 8
Close5Open4Quit5Save5Save as5Flat systems2Full description2Setting1Full description for customized object7GGeneral featuresCustomized objects7	1 9 1 0 0 4 0 8 7
Close5Open4Quit5Save5Save as5Flat systems2Full description2Setting1Full description for customized object7GGeneral featuresCustomized objects7Submodels7	191004 08 77
Close5Open4Quit5Save5Save as5Flat systems2Full description2Full description for customized object7GGGeneral features7Submodels7Supercomponent7	191004 08 779
Close5Open4Quit5Save5Save as5Flat systems2Full description2Setting1Full description for customized object7GGGeneral features7Submodels7Supercomponent7Generic objects7	191004 08 7792
Close5Open4Quit5Save5Save as5Flat systems2Full description2Setting1Full description for customized object7GGGeneral features7Submodels7Supercomponent7Generic objects7Generic objects7Gustomized objects7Supercomponent7Generic objects7Generic objects7Generic objects7Global parameters7	191004 08 7792
Close5Open4Quit5Save5Save as5Flat systems2Full description2Setting1Full description for customized object7GGGeneral features7Submodels7Supercomponent7Generic objects7Generic objects7Supercomponent7Supercomponent8Enumeration parameter8	191004 08 7792 7
Close5Open4Quit5Save5Save as5Flat systems2Full description2Setting1Full description for customized object7G6General features7Submodels7Supercomponent7Generic objects7Generic objects7Supercomponent7Supercomponent8Integer8Integer8	1 9 1 0 0 4 0 8 7 7 9 2 7 7
Close5Open4Quit5Save5Save as5Flat systems2Full description2Setting1Full description for customized object7G6General features7Submodels7Supercomponent7Generic objects7Generic objects8Integer8Real8	191004 08 7792 776
Close5Open4Quit5Save5Save as5Save as5Flat systems2Full description2Setting1Full description for customized object7G7General features7Submodels7Supercomponent7Generic objects7Global parameters8Integer8Real8Text8	1 9 1 0 0 4 0 8 7 7 9 2 7 7 6 8

H

Help menu	
About	'4
OnLine	'3
Hidden check boxes	1
Hide all	0
Hide status	
Right button menus	3
HTML documentation	
Create HTML skeleton7	'2
View HTML documentation7	'2
I	
- Icon designer	51
Icon for customized object	7
Icons menu	
Add category	;8
Add component	50
Icon designer 6	51
Remove category 5	;9
Remove component	51
Integer parameters 9	94
Internal items lists 9)2
Internal variables 9)3
	-
Last opened files list	1
List of constituents of customized supercomponent	50 10
List of currently opened files	3
Μ	
Main window	7
Menu bar	9
Move Up	62
Multi-levels customization	3
Ν	
Name conventions	7
Name of customized object 7	, 7
	'
0	
OnLine	3
Options menu	
AMECustom Preferences	7
Color preferences	6
Path list	5
Р	
Passwords	
Required to open encrypted object	6
Path list	6
Preview facilities	3

Preview menu	
Parameter mode	58
Run mode	58
R	
Real parameters	93
S	
Saving a customized object	12
Start AMESet	71
Start AMESim	71
Starting AMECustom	8
State count facility	
Hidden states	43
Submodel List with customized submodels	14
Supercomponents	
Listing	54
Removing	54
Т	
Text parameters	94
Title	
Right click menus	95
Toolbar	74
Tools menu	
License viewer	71
Trash can	53
T	
User submodels	
List of	55
Removing	55
V	
Value	
Right click menu 90.92 0	95
	, ,
W	
Windows menu	
Cascade	73

Reporting Bugs and using the Hotline Service

AMECustom is a large piece of software containing many hundreds of thousands of lines of code. With software of this size it is inevitable that it contains some bugs. Naturally we hope you do not encounter any of these but if you use **AMECustom** extensively at some stage, sooner or later, you may find a problem.

Bugs may occur in the pre- and post-processing facilities of **AMESim**, **AMERun**, **AMESet**, **AMECustom** or in one of the interfaces with other software. Usually it is quite clear when you have encountered a bug of this type.

Bugs can also occur when running a simulation of a model. Unfortunately it is not possible to say that, for any model, it is always possible to run a simulation. The integrators used in **AMECustom** are robust but no integrator can claim to be perfectly reliable. From the view point of an integrator, models vary enormously in their difficulty. Usually when there is a problem it is because the equations being solved are badly conditioned. This means that the solution is ill-defined. It is possible to write down sets of equations that have no solution. In such circumstances it is not surprising that the integrator is unsuccessful. Other sets of equations have very clearly defined solutions. Between these extremes there is a whole spectrum of problems. Some of these will be the marginal problems for the integrator.

If computers were able to do exact arithmetic with real numbers, these marginal problems would not create any difficulties. Unfortunately computers do real arithmetic to a limited accuracy and hence there will be times when the integrator will be forced to give up. Simulation is a skill which has to be learnt slowly. An experienced person will be aware that certain situations can create difficulties. Thus very small hydraulic volumes and very small masses subject to large forces can cause problems. The *State count* facility can be useful in identifying the cause of a slow simulation. An eigenvalue analysis can also be useful.

The author remembers spending many hours trying to understand why a simulation failed. Eventually he discovered that he had mistyped a parameter. A hydraulic motor size had been entered making the unit about as big as an ocean liner! When this parameter was corrected, the simulation ran fine.

It follows that you must spend some time investigating why a simulation runs slowly or fails completely. However, it is possible that you have discovered a bug in an **AMESIM** submodel or utility. If this is the case, we would like to know about it. By reporting problems you can help us make the product better.

On the next page is a form. When you wish to report a bug please photocopy this form and fill the copy. You telephone us, having the filled form in front of you means you have the information we need. Similarly include the information in an email.

To report the bug you have three options:

- reproduce the same information as an email
- telephone the details
- fax the form

Use the fax number, telephone number or email address of your local distributor.
HOTLINE REPORT

Creation date:	Created by:	
Company:	Contact:	
Keywords (at least one):		
Problem type:	🗆 Bug	□ Other
Summary:		
Description		
Description.		

Involved operating system(s):

🗆 Unix (all)	□ PC (all)
□ HP	☐ Windows 2000
□ ІВМ	□ Windows NT
□ SGI	□ Windows XP
	🗆 Linux
□ Other:	□ Other:

Involved software version(s):

□ AMESim (all)	□ AMERun (all)	□ AMESet (all)	□ AMECustom (all)
AMESim 4.0	AMERun 4.0	AMESet 4.0	□ AMECustom 4.0
AMESim 4.0.1	AMERun 4.0.1	□ AMESet 4.0.1	AMECustom 4.0.1
AMESim 4.0.2	AMERun 4.0.2	□ AMESet 4.0.2	AMECustom 4.0.2
AMESim 4.0.3	AMERun 4.0.3	AMESet 4.0.3	AMECustom 4.0.3
□ AMESim 4.1	AMERun 4.1	AMESet 4.1	□ AMECustom 4.1
AMESim 4.1.1	AMERun 4.1.1	□ AMESet 4.1.1	AMECustom 4.1.1
AMESim 4.1.2	AMERun 4.1.2	□ AMESet 4.1.2	AMECustom 4.1.2
□ AMESim 4.1.3	AMERun 4.1.3	□ AMESet 4.1.3	AMECustom 4.1.3
□ AMESim 4.2	AMERun 4.2	□ AMESet 4.2	□ AMECustom 4.2

Web Site http://www.amesim.com

Headquarter & Development Center

Tel: +33 4-77-23-60-30

Fax: +33 4-77-23-60-31 E-mail: <u>hotline@amesim.com</u>

FRANCE - SWITZERLAND -SPAIN - PORTUGAL - BENELUX

IMAGINE S.A. Paris

Tel: +33 1-39-43-08-12 Fax: +33 1-39-43-52-19 E-mail: <u>hotline@amesim.com</u>

ITALY - SWITZERLAND

IMAGINE S.A. Lyon

Tel: +33 4-37-69-72-30 Fax: +33 4-78-54-39-61 E-mail: <u>hotline@amesim.com</u>

Tel: +44 (0) 1869 351 994 Fax: +44 (0) 1869 351 302 E-mail: <u>support.uk@amesim.com</u>

GERMANY - AUSTRIA - FINLAND -DENMARK - NETHERLANDS -NORWAY - SWEDEN -SWITZERLAND - EASTERN EUROPE

Gine Software

Tel: +49 (0) 89 / 548495-35 Fax: +49 (0) 89 / 548495-11 E-Mail: <u>hotline.germany@amesim.com</u>

HUNGARY

Budapest University of Technology & Economics

Tel: (36) 1 463 4072 / 463 2464 Fax: (36) 1 463 3464 E-Mail: vad@simba.ara.bme.hu

NORTH AMERICA

IMAGINE Software, Inc.

Tel: (1) 734-207-5557 Fax: (1) 734-207-0117 E-Mail: <u>support.us@amesim.com</u>

SOUTH AMERICA

KEOHPS Ltd Tel: (55) 48 239 – 2281 Fax: (55) 48 239 – 2282 E-Mail: <u>info@keohps.com</u>

JAPAN



Tel : +81 (0) 3 3351 9691 Fax : +81 (0) 3 3351 9692 E-mail: <u>imagine-japan@amesim.com</u>

CHINA

Tel: + 86 21 34 12 34 58

E-mail: qing@amesim.com

United Right Technology

Tel: (86) 10-67082450(52)(53)(54) Fax: (86) 10-67082449 E-Mail: <u>urt@urtgroup.com</u>

SOUTH KOREA

SHINHO Systems Co., Ltd

Tel: +82 31 608 0434 Fax: +82 31 608 0439 E.Mail: <u>iclee@shinho-systems.co.kr</u>

ISRAEL

Tel : +972 3534 4432 Fax : +972 3535 5514 E-mail: <u>fempp@netvision.net.il</u>