

# COMSOL<sup>TM</sup> RELEASE NOTES

**VERSION 3.2 a**



**How to contact COMSOL:**

**Benelux**

COMSOL BV  
Röntgenlaan 19  
2719 DX Zoetermeer  
The Netherlands  
Phone: +31 (0) 79 363 4230  
Fax: +31 (0) 79 361 4212  
info@femlab.nl  
www.femlab.nl

**Denmark**

COMSOL A/S  
Diplomvej 376  
2800 Kgs. Lyngby  
Phone: +45 88 70 82 00  
Fax: +45 88 70 80 90  
info@comsol.dk  
www.comsol.dk

**Finland**

COMSOL OY  
Lauttasaarentie 52  
FIN-00200 Helsinki  
Phone: +358 9 2510 400  
Fax: +358 9 2510 4010  
info@comsol.fi  
www.comsol.fi

**France**

COMSOL France  
19 rue des bergers  
F-38000 Grenoble  
Phone: +33 (0)4 76 46 49 01  
Fax: +33 (0)4 76 46 07 42  
info@comsol.fr  
www.comsol.fr

**Germany**

FEMLAB GmbH  
Berliner Str. 4  
D-37073 Göttingen  
Phone: +49-551-99721-0  
Fax: +49-551-99721-29  
info@femlab.de  
www.femlab.de

**Italy**

COMSOL S.r.l.  
Contrada Santa Croce, 22  
25125 Brescia  
info.it@comsol.com

**Norway**

COMSOL AS  
Verftsgata 4  
NO-7485 Trondheim  
Phone: +47 73 84 24 00  
Fax: +47 73 84 24 01  
info@comsol.no  
www.comsol.no

**Sweden**

COMSOL AB  
Tegnérsgatan 23  
SE-111 40 Stockholm  
Phone: +46 8 412 95 00  
Fax: +46 8 412 95 10  
info@comsol.se  
www.comsol.se

**Switzerland**

FEMLAB GmbH  
Technoparkstrasse 1  
CH-8005 Zürich  
Phone: +41 (0)44 445 2140  
Fax: +41 (0)44 445 2141  
info@femlab.ch  
www.femlab.ch

**United Kingdom**

COMSOL Ltd.  
Studio G8 Shepherds Building  
Rockley Road  
London W14 0DA  
Phone: +44-(0)-20 7348 9000  
Fax: +44-(0)-20 7348 9020  
info.uk@comsol.com  
www.uk.comsol.com

**United States**

COMSOL, Inc.  
1 New England Executive Park  
Suite 350  
Burlington, MA 01803  
Phone: +1-781-273-3322  
Fax: +1-781-273-6603

COMSOL, Inc.  
1100 Glendon Avenue, 17th Floor  
Los Angeles, CA 90024  
Phone: +1-310-689-7250  
Fax: +1-310-689-7527

COMSOL, Inc.  
744 Cowper Street  
Palo Alto, CA 94301  
Tel: +1-650-324-9935  
Fax: +1-650-324-9936

info@comsol.com  
www.comsol.com

For a complete list of international  
representatives, visit  
[www.comsol.com/contact](http://www.comsol.com/contact)

**Company home page**

[www.comsol.com](http://www.comsol.com)

**COMSOL user forums**

[www.comsol.com/support/forums](http://www.comsol.com/support/forums)

*COMSOL 3.2a Release Notes*

© COPYRIGHT 1994–2005 by COMSOL AB. All rights reserved

Patent pending

The software described in this document is furnished under a license agreement. The software may be used or copied only under the terms of the license agreement. No part of this manual may be photocopied or reproduced in any form without prior written consent from COMSOL AB.

COMSOL, COMSOL Multiphysics, COMSOL Script, and COMSOL Reaction Engineering Lab are trademarks of COMSOL AB.

Other product or brand names are trademarks or registered trademarks of their respective holders.

Version: December 2005 COMSOL 3.2a

# C O N T E N T S

## Chapter 1: COMSOL 3.2a Release Notes

<b>Introduction</b>	<b>2</b>
Obtaining COMSOL 3.2a . . . . .	2
Installing COMSOL 3.2a . . . . .	2
System Requirements for 64-bit Windows Versions. . . . .	2
The COMSOL 3.2a Documentation Set . . . . .	3
COMSOL 3.2a Release Highlights . . . . .	3
The COMSOL Reaction Engineering Lab . . . . .	4
<b>New Features in COMSOL Multiphysics</b>	<b>5</b>
Creating a Geometry from a Mesh . . . . .	5
Frame Selection for Integration Coupling Variables . . . . .	6
Remeshing for Moving Mesh and Parameterized Geometries . . . . .	6
Geometry Repair When Creating Composite Objects. . . . .	12
Mesh Optimization in 2D. . . . .	12
Global Expression Variables . . . . .	13
Function for Numbering of Elements, Nodes, and DOFs . . . . .	13
Stop Condition for the Time-Dependent and Parametric Solvers. . . . .	15
Stop if Error Due To Undefined Operations. . . . .	15
Plotting Global Expressions . . . . .	15
Contour Labels . . . . .	17
Improved Streamline Plots and Particle Tracing . . . . .	17
Improved Scripting Support in COMSOL Script . . . . .	18
Texts Including HTML Formatting, Math Symbols, and More . . . . .	18
Improved Image Export . . . . .	19
Explicit View Settings . . . . .	20
Antialiasing and Line Width as Preference Settings . . . . .	21
Entering Draw Mode . . . . .	22
New Command-Line Features for Postprocessing . . . . .	23
Updated and Corrected Models . . . . .	24
<b>New Features in the Electromagnetics Module</b>	<b>26</b>
Higher-Order Vector Elements . . . . .	26

Vector Elements for All Mesh Element Types . . . . .	26
Multigrid Support in Electromagnetic Wave Models . . . . .	26
Multigrid Support for AV Quasi-Statics and Magnetostatics Models . . . . .	26
Transient Quasi-Statics, Vector Potential—Induction Currents. . . . .	26
Import of SPICE Netlists via COMSOL Script . . . . .	27
New Model—Inductance of a Power Inductor. . . . .	28
<b>New Features in the MEMS Module</b>	<b>29</b>
Predefined Multiphysics Couplings for Flow with Species Transport. . . . .	29
New Models . . . . .	29
<b>New Features in the Structural Mechanics Module</b>	<b>31</b>
The Truss Application Modes . . . . .	31
Updated and Corrected Models . . . . .	32
<b>New Features in COMSOL Script</b>	<b>34</b>
Improved Plots . . . . .	34
Support for Formatting and Symbols in Texts . . . . .	34
New Functions . . . . .	37
Updated and Improved Functions. . . . .	40
User-Defined Classes . . . . .	41
Encrypting M-files . . . . .	41
<b>Chemical Engineering Module Updates</b>	<b>42</b>
Updated Rising Bubble Model Using Level Sets . . . . .	42
<b>INDEX</b>	<b>43</b>

# COMSOL 3.2a Release Notes

These Release Notes provide information about new features and changes in the COMSOL 3.2a update. They are intended for users who receive COMSOL 3.2a as part of the COMSOL software maintenance program as well as for new users of the COMSOL 3.2a software products. These notes are a complement to the COMSOL 3.2 printed and online documentation. If you receive the COMSOL 3.2a CDs, this document is available in the root directory of CD 1.

# Introduction

Welcome to COMSOL 3.2a! These *Release Notes* bring you up to date on the new features in this software update.

## *Obtaining COMSOL 3.2a*

---

If you own a COMSOL license and are an active subscriber to the COMSOL maintenance program, you can download the COMSOL 3.2a software update from <http://www.comsol.com/support/updates/>.

If you are not an active subscriber to the COMSOL maintenance program or have any questions regarding the COMSOL 3.2a update, contact your COMSOL sales representative. For contact information on COMSOL's worldwide offices and representatives, see [www.comsol.com/contact/](http://www.comsol.com/contact/).

## *Installing COMSOL 3.2a*

---

If you download the COMSOL 3.2a update from the COMSOL web site, follow the directions provided on the web pages for the software update.

For installations of COMSOL 3.2a using the product CDs, you can find complete information about system requirements and installation procedures in the *COMSOL Installation Guide*.

If you change the license or do a new installation using a Named Single User License (NSL), the installer prompts you for the username to bind to the license and then creates the required options file for you.

## *System Requirements for 64-bit Windows Versions*

---

- Windows XP Professional x64 Edition
- A PC with one of these processors: AMD Opteron, AMD Athlon 64, Pentium 4 with EM64T, or Xeon with EM64T
- A graphics card with at least 32 MB of memory
- At least 256 MB of system memory

## *The COMSOL 3.2a Documentation Set*

---

These *Release Notes* are part of the documentation set for COMSOL 3.2a and are available for download from the COMSOL web site as part of the COMSOL 3.2a update.

A new product with this release, the COMSOL Reaction Engineering Lab, comes with the *COMSOL Reaction Engineering Lab User's Guide*, a printed document with comprehensive information about the software and its applications within chemical reaction engineering.

All other documentation is identical to the COMSOL 3.2 documentation.

## *COMSOL 3.2a Release Highlights*

---

The COMSOL 3.2a release includes the following new features in the COMSOL software products:

- 64-bit Windows versions of the COMSOL software products for Windows XP Professional x64 Edition.
- A new software package for chemical reaction engineering, the COMSOL Reaction Engineering Lab. It provides full support for the modeling, simulation, and visualization of chemical reactions and provides a seamless interface to the Chemical Engineering Module.
- The ability to use a mesh or a deformed mesh to define a model geometry.
- Remeshing capabilities in COMSOL Multiphysics for improved modeling of moving boundaries and parameterized geometries.
- Support for both reference frames and spatial frames in the dialog box for integration coupling variables. This feature applies to models that include Moving Mesh (ALE) or Parameterized Geometry application modes.
- Improved rendering and graphics in COMSOL Multiphysics and COMSOL Script plot windows. These features include contour labels as well as support for Greek letters, mathematical symbols, and Unicode characters in plot titles, axis labels, legends, and other similar plot elements.
- Improved image export.
- Global expression variables for defining expressions that are valid across all geometries as well as in ODEs and algebraic equations in COMSOL Multiphysics.

- New features in the Electromagnetics Module:
  - Higher-order vector elements
  - Vector elements available for all mesh types
  - Transient analysis in 3D models using the application mode for induction currents using quasi-statics with a magnetic vector formulation
  - Predefined settings and explicit gauge fixing for effective use of the multigrid solvers in magnetostatics and quasi-statics
  - Import of SPICE netlists via COMSOL Script
- New truss/cable elements in the Structural Mechanics Module.
- New predefined multiphysics couplings for flow with species transport in the MEMS Module.
- Support for user-defined objects in COMSOL Script as well as many new functions for image import, polynomials, graphics, operating system commands, and more.
- Several new and updated models.
- Many small improvements and corrections.

### *The COMSOL Reaction Engineering Lab*

---

The COMSOL Reaction Engineering Lab is a new software product for the modeling and simulation of chemical systems. The software automatically generates the chemical kinetics of a reaction process based on reaction formulas you enter. You can add and remove reaction steps or modify rate equations to explore how they affect a reactor's performance

#### **SEAMLESS INTERFACE TO THE CHEMICAL ENGINEERING MODULE**

You can extend any model in the COMSOL Reaction Engineering Lab to include a real-world geometry and multiphysics couplings. It is easy to export the mass balances, heat balances, and momentum balances into corresponding application modes in the Chemical Engineering Module.

For more information about the COMSOL Reaction Engineering Lab, visit [www.comsol.com/products/reaction](http://www.comsol.com/products/reaction) or contact your local COMSOL representative.

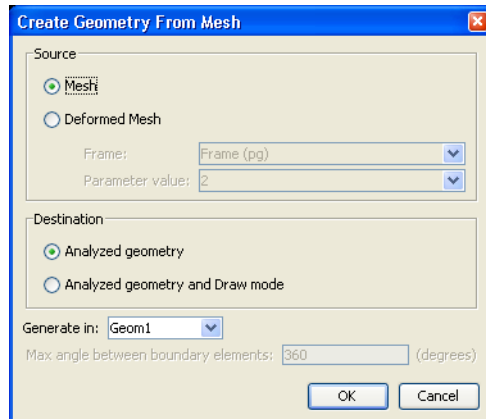
# New Features in COMSOL Multiphysics

The following section describes new and improved features in COMSOL Multiphysics.

## *Creating a Geometry from a Mesh*

---

When you import a mesh there is no corresponding geometry. A similar situation occurs when you have created a deformed mesh and want to remesh using the deformed geometry. COMSOL Multiphysics 3.2a provides the ability to create a geometry from such meshes. This *analyzed geometry* is the geometry that you use for modeling. You can also use the created geometry in Draw mode. To generate a geometry from a mesh, choose **Create Geometry from Mesh** from the **Mesh** menu.



In the **Source** area, select if you want to create a geometry from a mesh or if you want to create a geometry from a deformed mesh, which is retrieved from solving a parameterized geometry or a moving mesh (ALE) problem. If you want to create a geometry from a deformed mesh, you also need to specify which frame and solution to use.

In the **Destination** area, select the **Analyzed geometry** option to create only an analyzed geometry, or select the **Analyzed geometry and Draw mode** option to create both an analyzed geometry and a Draw mode object from the mesh. If you select **Deformed mesh** as the source, these options also generate a new mesh, and the destination options are **Mesh and analyzed geometry** and **Mesh, analyzed geometry, and Draw mode**.

You can select from the **Generate in** list in which geometry the software places the generated analyzed geometry. The default destination is the current geometry, but you can also choose from any other geometries in the model or create a new geometry with the generated analyzed geometry.

In the **Max angle between boundary elements** edit field you specify the maximum allowed angle between two boundary elements as part of the same face. This parameter is available only if the mesh has no parameterization.

---

**Note:** When you generate an analyzed geometry from a 3D mesh and the **Max angle between boundary elements** parameter is available, the generation may introduce new boundaries.

---

The equivalent command-line function is `mesh2geom`. Type

```
help mesh2geom
```

for information about the syntax and an example.

See “Remeshing Explained With an Example” on page 7 for an example where you create a new geometry from a deformed mesh.

### *Frame Selection for Integration Coupling Variables*

---

When working with integration coupling variables and the model includes multiple frames, you can select the appropriate frame from a list on the **Source** tab. The frame determines the volume measure to use when computing the integral.

### *Remeshing for Moving Mesh and Parameterized Geometries*

---

When you work with the Moving Mesh or Parameterized Geometry application modes, the mesh quality becomes poor when the deformations are large. This section describes how to remesh the geometry after a part of the solution steps and how to continue solving.

## **SMOOTHING METHODS AND COMPATIBILITY WITH 3.2 MODELS**

The Moving Mesh application mode has two smoothing methods, Laplace smoothing and Winslow smoothing, which are responsible for creating a smooth deformed mesh. We have reimplemented Winslow smoothing in 3.2a to work with the remeshing. This new implementation uses mesh positions instead of mesh displacements as degrees of

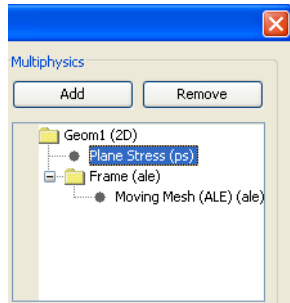
freedom. For backward compatibility, 3.2 models that use Winslow smoothing retain the old implementation. You can change the choice of smoothing method in the **Application Mode Properties** dialog box.

The Parameterized Geometry application mode also includes both smoothing methods.

### REMESHING EXPLAINED WITH AN EXAMPLE

The section describes the remeshing technique by building a simple model where a force bends a beam.

- 1 In the **Model Navigator** select **2D** from the **Space dimensions** list.
- 2 In the **COMSOL Multiphysics** folder, select **Deformed Mesh>Moving Mesh (ALE)>Transient analysis**.
- 3 Click the **Multiphysics** button and then the **Add** button to add the Moving Mesh application mode to the model.
- 4 In the list of application modes, select **Structural Mechanics>Plane Stress>Transient analysis**. In the **Multiphysics** area select the node **Geom 1 (2D)**. Then click **Add** to add the Plane Stress application mode.



- 5 Click **OK** to close the **Model Navigator**.

#### *Geometry Modeling*

- 1 Draw a rectangle with a width of 1.6, height of 1.2, and with its lower left corner at  $(-0.6, -0.4)$ .
- 2 Draw a second rectangle with a width of 1.4, height of 0.2, and its lower left corner at  $(-0.6, -0.2)$ .

#### *Physics Settings*

- 1 In the **Subdomain Settings** dialog box of the Plane Stress application mode clear the **Active in this domain** check box for subdomain 1. Click **OK** to close the dialog box.

- 2 In the **Boundary Settings** dialog box select boundary 3 and activate the constraints  $R_x$  and  $R_y$  by selecting the corresponding check boxes. Go to the **Load** tab and select boundaries 4, 6, and 8. Enter a force that increases with time by setting  $F_y$  to  $-1e6*t$ . Click **OK** to close the dialog box.
- 3 Select the **Moving Mesh (ALE)** application mode from the **Multiphysics** menu.
- 4 In the **Subdomain Settings** dialog box of the **Moving Mesh** application mode select subdomain 2 and click the **Physics induced displacement** button. Enter the displacement variables  $u$  and  $v$ . Click **OK** to close the dialog box.
- 5 In the **Boundary Settings** dialog box select boundaries 4, 6, and 8. Select the **dx** and **dy** check boxes and then enter the mesh displacement  $u$  and  $v$  in the corresponding edit fields.
- 6 Select boundaries 1, 2, 5, 7, and 9, and select both the **dx** and **dy** check boxes to prevent these boundaries from moving. Click **OK** to close the dialog box.

Note that to obtain a correct solution, you should use the support for large deformations in the Plane Stress application mode, but that feature is available only in the extended Plane Stress application mode in the Structural Mechanics and MEMS modules.

The next step is to activate support for remeshing in the **Moving Mesh** application mode:

- 1 Open the **Application Mode Properties** dialog box by selecting **Properties** from the **Physics** menu.
- 2 Select **On** in the **Allow remeshing** list and click **OK**.

You must make this selection before solving the model. If you attempt to solve it and later realize that you need to remesh while remeshing support is not activate, you must go back, activate it, and re-solve the model.

#### *Mesh Generation*

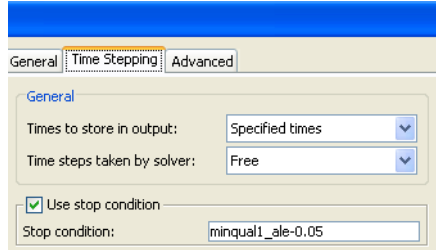
- 1 Initialize the mesh.
- 2 Refine it once.

#### *Computing the Solution*

In order to make the solver stop when the mesh quality becomes poor, you must enable a stop condition in the **Solver Parameters** dialog box.

- 1 From the **Solve** menu open the **Solver Parameters** dialog box. In the **Times** edit field enter  $0:0.01:0.3$ .

- 2 Go to the **Time Stepping** tab and select the **Use stop condition** check box.



The solver halts when the expression in the **Stop condition** edit field becomes negative. The default value for the stop condition is `minqua1_ale-0.05`. The Moving Mesh application mode defines the variable `minqua1_ale`, which represents the minimum quality of the deformed mesh. The variable name begins with `minqua1` followed by the geometry number. This nomenclature creates unique variables in the case where a model has multiple geometries. For models with one geometry this number is 1, resulting in `minqua1`. The variable name for the minimum quality ends with an underscore (`_`) followed by the name of the frame for the deformed mesh. The name of the frame is `ale`, which means that the name of the quality variable becomes `minqua1_ale`. To find the name of the frame, open the **Model Navigator** and see to which frame the Moving Mesh application mode is attached. You can also look in the **Application Mode Properties** dialog box. The **Defines frame** property gives the name of the frame of the deformed mesh.

The default stop condition dictates that the solver should stop when the minimum quality is less than 0.05. Depending on the initial quality of the mesh, you might have to change this number.

If the model has more than one geometry with a deformed mesh, the stop condition uses the minimum quality of all the geometries. In a model with two geometries, use the stop condition `min(minqua1_ale,minqua2_ale)-0.05`.

- 3 Click **OK** to close the **Solver Parameters** dialog box and then click the **Solve** button on the Main toolbar.

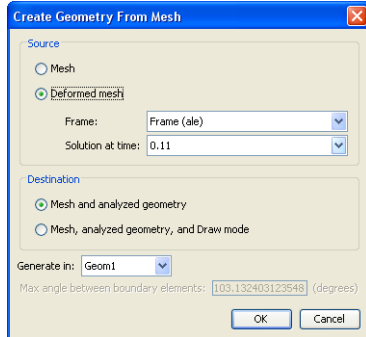
The solver stops somewhere around  $t = 0.12$ .

### *Geometry Modeling*

Next you need to create a new geometry and mesh from the deformed mesh.

- 1 From the **Mesh** menu open the **Create Geometry From Mesh** dialog box.

- 2 Click the **Deformed mesh** button and then **OK**.



This creates a new geometry from the deformed mesh at the last time step.

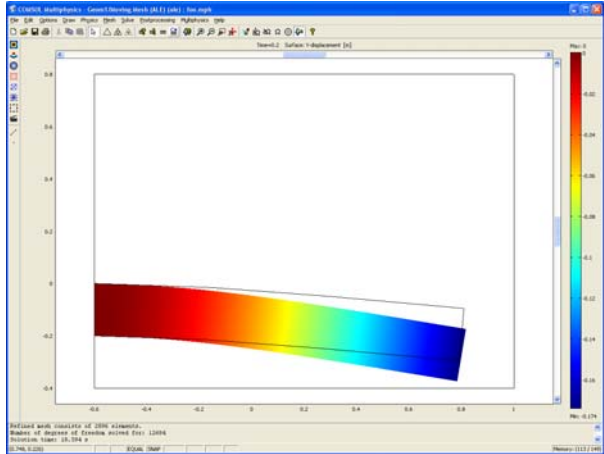
### *Mesh Generation*

Using buttons on the Main toolbar, initialize the mesh and refine it once. This creates a new mesh with higher quality for the deformed geometry.

### *Computing the Solution*

- 1 Open the **Solve>Solver Parameters** dialog box and on the **General** tab change the list of times in the **Times** edit field to start from the last time of the current solution. Generally when solving time-dependent problems this is not absolutely necessary, but in this case it is important because the force in the Plane Stress application mode explicitly depends on the time.
- 2 When you continue solving the model, it is important to use the solution from the last time step as the new initial value when restarting the solver. Do this by clicking the **Restart** toolbar button.

The solver continues solving and stops when the mesh quality again becomes poor. To postprocess the two parts of the simulation together, you can export the FEM structure to the command line after each part of the simulation. Use **File>Export>Export FEM Structure** to save each FEM structure with a new name. Then use the new ability of the `postmove` command to postprocess several FEM structures (see “Animation Using Multiple FEM Structures” on page 23).



### RESTORING THE ORIGINAL GEOMETRY

Assume you want to go back and restart the solver from the very beginning. In this case you first must restore the original geometry. To do so, go to the Draw mode. Because you have generated a deformed geometry, the objects in Draw mode do not represent the current geometry. The software therefore asks if you want to use the current objects in Draw mode or replace them with the deformed geometry (see “Entering Draw Mode” on page 22). Using the current Draw mode objects restores the original geometry.

### FRAMES WHEN USING REMESHING

When you set the application mode property **Allow remeshing to On**, the software adds an extra frame. The three frames are now:

- The *spatial frame*, usually denoted by `a1e` or `pg` (the name of the application mode controlling the deformed mesh). This is the usual coordinate system. The corresponding coordinates are called `x` and `y` by default.
- The *reference frame*, usually denoted by `ref`. This coordinate system describes the original configuration (before remeshing). The corresponding coordinates are called `X` and `Y` by default. If the mesh movement follows the material movement (as is common in solid mechanics), this frame is also called the *material frame*. In this case, `X` and `Y` are the coordinates of the material point in the original configuration. The software stores the values of `X` and `Y` for all nodes as degrees of freedom in the

solution vector but never solves for them. Their values only change when mapping the solution, which happens when you click the **Restart** button after a remeshing.

- The *mesh frame*, usually denoted by **mesh**. This coordinate system describes the configuration just after the latest remeshing. The corresponding coordinates are called  $X_m$  and  $Y_m$  by default.

### LINEAR VS. QUADRATIC ELEMENTS

When using second-order (or higher-order) elements in the deformed mesh application modes, the mesh moving techniques often produce elements with distorted shapes. The measure of mesh quality does not capture these distorted shapes because it is computed from the positions of the corners of the mesh element (ignoring mid-side nodes, for instance). For these reasons, it is often best to use linear elements for the mesh positions in the deformed mesh application modes. You can select element type on the **Element** tab in the **Subdomain Settings** dialog box for the Moving Mesh and Parametrized Geometry application mode. You must also open the **Model Settings** dialog box from the **Physics** menu and select the ALE frame (typically **Frame (ale)**) in the **Frames** list and select the appropriate element order (typically **Linear**) in the **Geometry shape order** list.

### *Geometry Repair When Creating Composite Objects*

---

You can now supply a repair tolerance when creating composite objects. The software uses this tolerance to remove small entities and to heal gaps in the generated object. This can greatly improve the handling of imported CAD assemblies. To use this feature, open the **Create Composite Object** dialog box, select the **Repair** check box, then enter a value in the **Repair tolerance** edit field.

### *Mesh Optimization in 2D*

---

Mesh optimization is now also available in 2D for unstructured triangular meshes. The **Optimize quality** check box on the **Global** tab in the **Mesh Parameters** dialog box controls if COMSOL Multiphysics carries out quality optimization during mesh initialization. The default is to use optimizing of the mesh quality by smoothing the mesh. The smoother that the mesh generator uses when creating meshes improves the mesh quality by jiggling the mesh vertices, that is, by moving each mesh vertex toward the centerpoint of the surrounding mesh vertices and by performing edge swapping.

In COMSOL Script or MATLAB, you can use the corresponding function `meshsmooth` to select among different smoothing techniques and control the smoothing parameters.

### *Global Expression Variables*

---

*Global expression variables* are a new type of expression variables. Global expressions are available globally in the model across all geometries (including a 0D “ODE geometry” if you have added extra states or other degrees of freedom in the **ODE Settings** dialog box). They are otherwise similar to scalar expressions variables, which are available only in the current geometry.

To add or edit a global expression variable, go to the **Options** menu, point to **Expressions**, and then click **Global Expressions**.

### *Function for Numbering of Elements, Nodes, and DOFs*

---

The new function `xmeshinfo` (available in COMSOL Script and MATLAB when you use COMSOL Multiphysics) makes it possible to view information about the extended mesh. Use it to get information about node numbers, node coordinates, degree-of-freedom (DOF) numbers, element numbers, the number of DOFs, and more. The output from `xmeshinfo` contains information that conforms to the data structures in the mesh. Type

```
help xmeshinfo
```

for full information about available properties.

#### **NUMBERING CONVENTIONS**

The numbering provided by `xmeshinfo` corresponds to the numbering in the mesh data structure (see `femmesh`). The extended mesh uses a different numbering internally. All numbering is 1-based.

- **Elements.** For each mesh element type, `xmeshinfo` uses the element numbering in `femmesh`.
- **Node points.** The node points in `femmesh` have the same numbers in the extended mesh. Additional node points have higher numbers (these are arbitrarily ordered).
- **Local node points.** The numbering of local node points within a mesh element is different from the numbering in `femmesh`. However, both functions use the same definition of the local coordinate system. In the extended mesh, the local node points are in lexicographical order of their local coordinates. In `femmesh`, the mesh

vertices come first, in lexicographical order, and then come the other node points in lexicographical order (the latter are only present for a second-order mesh).

- **DOFs.** By default, the DOF number is the index in the complete set of degrees of freedom of the model. If you provide a value for the property `So1comp`, the DOF number is the index in the set of DOFs solved for. If you provide a value for the property `Null`, it is assumed that the Eliminate constraint-handling method is used, and the DOF number is the index in the set of unconstrained DOFs. This assumes a simple form of the constraints where each constraint constrains only one DOF. In other words, each column of the `Null` matrix has a single nonzero element. If `Null` does not have this form, an error message appears. The `Null` matrix is an output from the solvers (see `femlin`).

#### EXAMPLE OF USING XMeshInfo

Assume that `fem.mesh` is an imported NASTRAN mesh with second-order tetrahedral mesh elements, where node point numbering starts at 1. Use second-order Lagrange elements in the COMSOL Multiphysics model:

```
m = meshimport('nastrandemo1.nas');
fem.mesh = m{1};
fem.dim = 'u';
fem.shape = 2;
fem.equ.c = 1;
fem.bnd.h = 1;
fem.xmesh = meshextend(fem);
```

To get the DOF number corresponding to node point number 22 in the NASTRAN mesh, type

```
nodes = xmeshinfo(fem,'out','nodes');
nodes.dofs(1,22)
```

Compute an eliminated stiffness matrix and a null-space matrix with the line

```
[Kc,Null]=femlin(fem,'out',{'Kc' 'Null'});
```

To find the node point number corresponding to column 30 of `Kc` and its coordinates, type

```
dofs = xmeshinfo(fem,'out','dofs','null',Null);
n = dofs.nodes(30)
dofs.coords(:,30) % alternatively: nodes.coords(:,n)
```

To find the six DOF numbers in tetrahedron element 10 of the mesh, type

```
elements = xmeshinfo(fem,'out','elements','meshtype','tet2');
elements.dofs(:,10)
```

To find the total number of DOFs on the boundary, type

```
xmeshinfo(fem,'out','ndofs','edim',2)
```

The output from this function call reveals that there are 33,702 DOFs on the boundary.

### *Stop Condition for the Time-Dependent and Parametric Solvers*

---

On the **Time Stepping** and **Parametric** tabs in the **Solver Parameters** dialog box there is now a **Use stop condition** check box. If you select this check box, the solver stops before the expression in the **Stop condition** edit field becomes negative (the solver does not try to resolve the exact time for the zero crossing). The expression must be a scalar value that has no variation in space, typically a coupling variable with a global destination. When using a deformed mesh with remeshing, the Moving Mesh application mode provides a default expression that is the minimum quality of the deformed mesh minus 0.05 (see “Computing the Solution” on page 8).

### *Stop if Error Due To Undefined Operations*

---

By default, the solver stops with an error message when it encounters an undefined mathematical operation in an expression that appears in the model settings, for instance, division by zero or square root of a negative number. To change this behavior, go to the **Advanced** tab of the **Solver Parameters** dialog box and clear the **Stop if error due to undefined operation** check box. Then the solver treats the result of the operation as Inf (infinity) or NaN (not a number). This feature can be useful in a nonlinear problem where the steps in the iterative solution process lead to variable values for which an expression is undefined. The solver then reduces the step size in the Newton iteration when it encounters Inf or NaN so that it can find a solution. The corresponding property for the command-line solver functions is `matherr on/off`.

### *Plotting Global Expressions*

---

To visualize globally defined variables such as solutions to ODEs, use the **Global Variables Plot** dialog box.

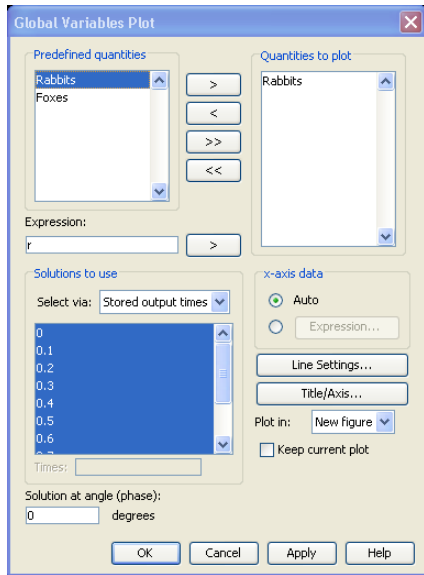


Figure 1-1: The Global Variables Plot dialog box.

The **Predefined quantities** list includes the names of all ODE variables (defined in the **ODE Settings** dialog box) and global expressions (defined in the **Global Expressions** dialog box). To plot one or more of these, select them and click the **>** button to add them to the **Quantities to plot** list. When you select a name in the **Predefined quantities** list, the software displays its variable name in the **Expression** edit field below the list. You can edit this name and add the edited quantity using the **>** button to the right of the text field. To remove one or more expressions from the **Quantities to plot** list, select them and click the **<** button. You can use Shift-click and Control-click in both lists.

In the **Solutions to use** area, select which time steps to include in the plot. By choosing **Interpolated times** in the **Select via** list, you can enter any vector of times in the **Times** edit field, such as `linspace(0,1,100)` or `0:0.02:1`.

Control the quantity to plot on the *x*-axis in the **x-axis data** area. If **Auto** is selected, the *x*-axis corresponds to time.

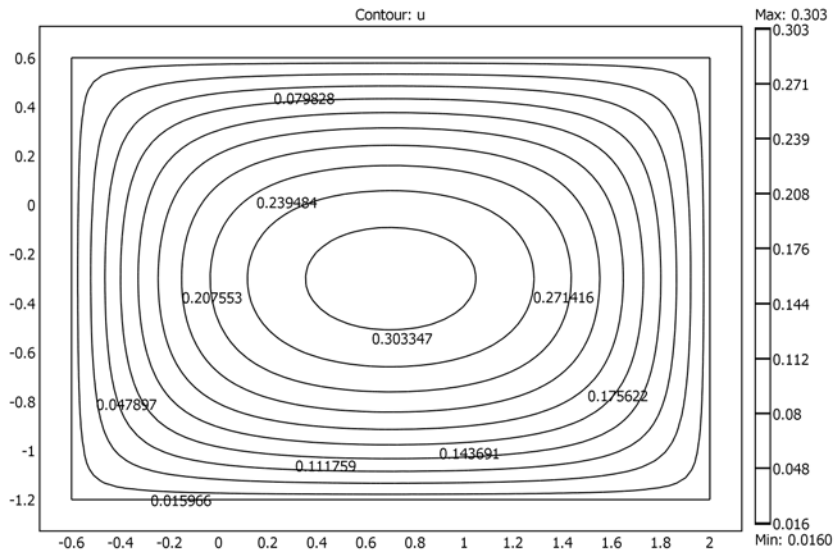
The other settings in this dialog box are similar to those in the **Cross-Section Plot Parameters** and **Domain Plot Parameters** dialog boxes (see Chapter 7, “Postprocessing and Visualization,” in the *COMSOL Multiphysics User’s Guide*)

To plot global variables in a script or at the command line, use the corresponding function `postglobalplot`.

### *Contour Labels*

---

You can now use labels that display the values of contours (isolevels). To include contour labels in a contour plot, open the **Plot Parameters** dialog box, click the **Contour** tab, and select the **Labels** check box in the **Contour levels** area.



*Figure 1-2: Contour labels in a plot using ten contour levels.*

### *Improved Streamline Plots and Particle Tracing*

---

COMSOL Multiphysics 3.2a features reorganized **Streamline** and **Particle Tracing** tabs in the **Plot Parameters** dialog box. The **Streamline** tab is only for stationary fields, and the **Particle Tracing** tab now includes an option to use either massless particles or particles with mass. Select which to use in the **Particle type** list and then click the **With Mass** or **Massless** tab to specify the equation of motion.

In addition to global coordinates, you can now use boundary coordinates to select start points on boundaries for the particles. On the **Particle Tracing** tab, click the option

button to the left of the **Boundary Coordinates** button; then click that button to open the **Boundary Coordinates** dialog box. Select the boundaries where you want your starting points. Click the **Number of points** button to use a number of equidistant points. Click the **Vector with boundary parameters** button to enter a vector of values from 0 to 1 to describe the start points. From the command line or in a script, use the `postcoord` function, which is the corresponding command-line function for computing the start points. The general plot function `postplot` can also call `postcoord` to compute the start points. Use the `postpart` property in `postplot` to pass a cell array with property/value pairs to `postcoord`.

### *Improved Scripting Support in COMSOL Script*

---

The functions in the following table are now available in COMSOL Script:

FUNCTION	DESCRIPTION
<code>flcontour2mesh</code>	Create boundary mesh from contour data
<code>flim2curve</code>	Create curve2 object from image data
<code>flmesh2spline</code>	Create spline curves from the mesh
<code>geomspline</code>	Spline interpolation
<code>helix2</code>	Create face helix in 3D
<code>helix3</code>	Create solid helix in 3D
<code>loft</code>	Loft 2D geometry sections to 3D geometry object

These additions mean that code using these functions work in COMSOL Script as well as MATLAB, and notes in the documentation that state you must run COMSOL Multiphysics with MATLAB no longer apply. For more information about these functions, use the command-line help in COMSOL Script or see the *COMSOL Multiphysics Command Reference*.

### *Texts Including HTML Formatting, Math Symbols, and More*

---

Text you add to plots using COMSOL Script can now include HTML tags, Greek letters, and mathematical symbols. You can also display other characters using the corresponding Unicode numbers. These options are also available for titles of postprocessing plots in COMSOL Multiphysics. For more information about these options, see “Support for Formatting and Symbols in Texts” on page 34.

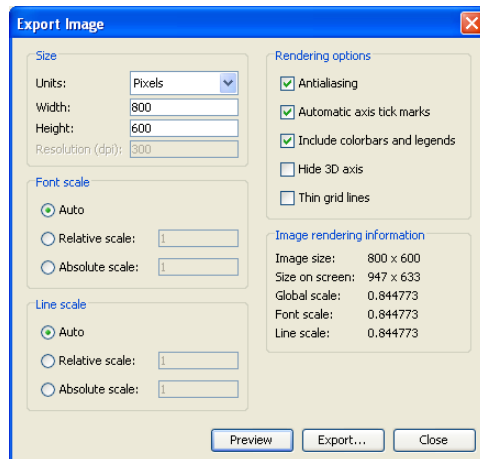
Please note several other smaller improvements:

- Using custom fonts and custom font sizes now works in colorbars and legends
- x-axis labels can now print to the right or left of the axis panel
- Major and minor tick marks are available when using log scales

### *Improved Image Export*

---

Using the new and improved **Export Image** dialog box, you can now control font sizes, line widths, and what to include in an exported image. To speed up the image-generation process, a preview feature and image rendering information are available.



*Figure 1-3: The Export Image dialog box.*

## **SETTINGS IN THE EXPORT IMAGE DIALOG BOX**

### *Scaling*

Settings in the **Font scale** and **Line scale** areas affect the scaling between the plot's size on the screen and the size in the image (size = number of pixels):

- Click **Auto** to use the global scale (you see its value in the **Image rendering information** area; also see “The Global Scale” on page 20) if you specify the size in pixels (the software scales text, lines, and other graphics equally). If you specify the size in centimeters or inches, the automatic scale is based on the resolution that you select. The font size and line width you specify when creating the plot are preserved if you

export an image using a certain resolution in dpi (dots per inch) and import it to a document as an image using the same dpi resolution (a text with a certain size in the plot looks like a text with the same size in the document).

- Click **Relative scale** to use a total font scale that is the automatic scale times the relative scale you specify.
- Click **Absolute scale** to use a total font scale that is equal to the absolute scale you specify.

## RENDERING OPTIONS

The following settings are available in the **Rendering options** area:

### *Antialiasing*

Select the **Antialiasing** check box to reduce stairstep-like lines (jaggies) and makes lines and edges look smooth.

### *Automatic Tick Marks*

Select the **Automatic tick marks** check box to take advantage of a new feature that can hide axis tick marks if they overlap. Clear this check box if you want make sure that the image has the same axis tick marks as you see on the screen.

### *Hiding the 3D Axis*

Select the **Hide 3D axis** check box to exclude the grid and coordinate system from the image.

### *Rendering Thin Grid Lines*

Select the **Thin grid lines** check box to render thin grid lines compared to other lines in the image. Use this option if you think that the grid is too dominating in the image.

## THE GLOBAL SCALE

The global scale is the scale between the size of the plot on screen and the size of the image (size = number of pixels).

### *Explicit View Settings*

---

You can now accurately control camera parameters such as the position, target, and view angle. To do so, open the **Visualization/Selection** dialog box from the **Options** menu and click the **Camera** tab.

The view settings for the camera parameters are available only in 3D.

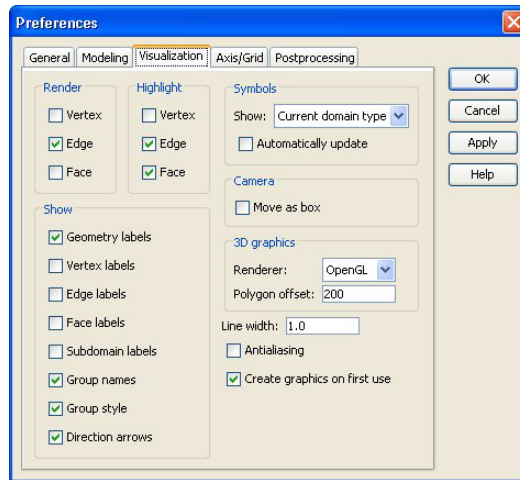
The corresponding command-line functions (`campos`, `camtarget`, `camva`, `camup`, and `camproj`) are available in COMSOL Script.

### *Antialiasing and Line Width as Preference Settings*

---

It is now possible to use antialiasing (smoothing graphics) when rendering in COMSOL Multiphysics. You can also change the line width.

To specify antialiasing and the line width, open the **Preferences** dialog box. Click the **Visualization** tab to access these properties.



*Figure 1-4: The Visualization tab in the Preferences dialog box.*

#### **ANTIALIASING**

To use antialiasing, select the **Antialiasing** check box. Antialiasing reduces the rendering speed, so the default setting is to not use antialiasing.

#### **LINE WIDTH**

Specify the line width as a scalar number in the **Line width** edit field. Figure 1-5 shows the effect of increasing the line width.

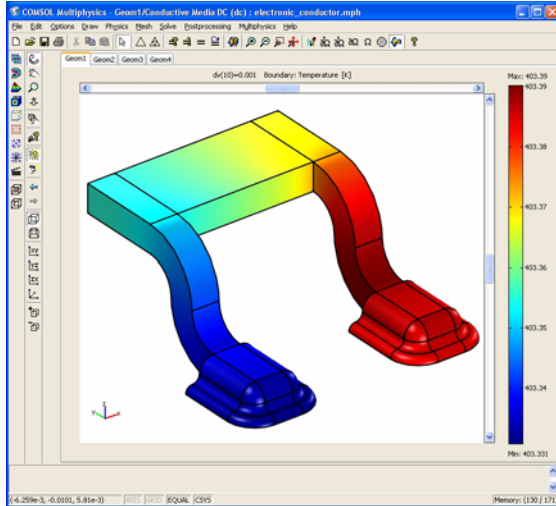
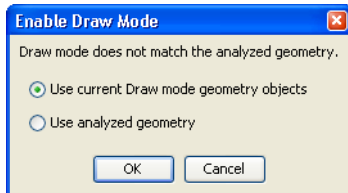


Figure 1-5: The effect of increasing line width.

### Entering Draw Mode

In certain situations, the Draw mode contents does not match the current analyzed geometry. This happens, for instance, when you replace the geometry and mesh with a deformed mesh so that the Draw mode still contains the original geometry while the analyzed geometry is the deformed geometry.

If you try to enter Draw mode when it does not match the analyzed geometry, the software displays a dialog box where you can choose between keeping the geometry objects in Draw mode or replacing them with the analyzed geometry. Then click **OK** to enter Draw mode.



If there is no analyzed geometry, as when working with an imported mesh, the option to use the analyzed geometry is not available. In that case you can use the **Create**

**Geometry From Mesh** dialog box in the **Mesh** menu to create an analyzed geometry and a Draw mode object from the imported mesh.

---

**Note:** If you open Draw mode using the **Use current Draw mode geometry objects** option, this action invalidates the old geometry, mesh, and solution.

---

### *New Command-Line Features for Postprocessing*

---

#### **NEW FUNCTION: POSTGLOBALPLOT FOR EVALUATING ODE VARIABLES**

The new function `postglobaleval` evaluates the ODE variables in `fem.ode.dim` and puts the results in a structure with the fields `x`, `y`, and `legend`. Here `x` is an array that contains the time steps, and `y` is a matrix that contains the values of all ODE variables where each column corresponds to an ODE variable. Further, `legend` is a cell array of legend strings. You can use `postglobaleval` to create data for plotting the solution to ODEs using the following code snippet if the FEM structure `fem` contains the solution to the ODEs:

```
data = postglobaleval(fem);
plot(data.x, data.y)
legend(data.legend)
```

#### **IMPROVED SELECTION OF EVALUATION POINTS IN POSTEVAL**

Use the new properties `Spoint`, `Bpoint`, and `Prpoint` in `posteval` to specify arbitrary local element evaluation points for simplex elements (triangular, tetrahedral, and edge elements), quadrilateral/block elements, and prism elements, respectively. If you specify any of these properties, the fields `t` and `q` in the output structure are empty, and the property `Cont` is neglected. To evaluate in Gauss points, use the new function `postgp` which returns Gauss points and Gauss point weights. You can use the output from `postgp` as input to the properties `Spoint`, `Bpoint`, and `Prpoint` in `posteval`.

The output structure from `posteval` now also includes the field `elind` with indices to the mesh element for each point.

#### **ANIMATION USING MULTIPLE FEM STRUCTURES**

The function `postmovie` can now take a cell array of FEM structures as input. This makes it possible to make animations of models using manual remeshing. For example, when you use the property `stopcond` in a solver in the user interface and then export

the FEM structure after each stop with the names fem1, fem2, and so on, you can make an animation of the entire model using the following command:

```
postmovie({fem1,fem2,fem3,...}, ...)
```

### *Updated and Corrected Models*

---

Updated models are available with the COMSOL Multiphysics 3.2a update on the COMSOL web site at <http://www.comsol.com/support/updates/> and in the updates folder on the COMSOL 3.2a CD 4.

#### **ELECTRONIC CONDUCTOR**

This introductory model, which appears in the *COMSOL Multiphysics Quick Start and Quick Reference*, includes a geometry file, `new_soldering_geom.mphbin`, which you can use if you want to skip the geometry modeling steps. This file was missing in COMSOL 3.2 but is now available for download from the COMSOL web site.

#### *Documentation Erratum*

Step 21 on page 62 in the *COMSOL Multiphysics Quick Start and Quick Reference* should say:

- In the **Name** edit field enter `air`.

#### **EIGENMODES OF A ROOM**

The eigenvalue formulation in the Acoustics application mode has changed from first-order eigenvalues to second-order eigenvalues. To get the same eigenfrequencies in the analysis, use another setting in the **Search for eigenvalues around** edit field. Replace Step 2 in “Computing the Solution” on page 26 in the *COMSOL Multiphysics Model Library* with the following:

- The frequency 100 Hz corresponds to  $-i\omega = -i2\pi \cdot 100 = -i628$ . Enter 15 in the **Desired number of eigenvalues** edit field and `-i1e2` in the **Search for eigenvalues around** edit field.

#### **SHALLOW WATER EQUATIONS**

The updated Shallow Water Equations model uses artificial stabilization and tighter tolerances for the time stepping. Full model documentation is available at <http://www.comsol.com/support/updates/> and in the updates folder on the COMSOL 3.2a CD 4.

### **SLOSHING TANK**

The updated Sloshing Tank model uses correct constraints for the free surface. Full model documentation is available at <http://www.comsol.com/support/updates/> and in the **updates** folder on the COMSOL 3.2a CD 4.

### **PACEMAKER ELECTRODE**

The following changes to the step-by-step instructions correct the geometry modeling for the Pacemaker Electrode model.

#### *Geometry Modeling*

- After Step 1 on page 67 in the *COMSOL Multiphysics Model Library*, add the following step:
- Click the **y-z** button and then click **OK**.
- In Step 7 on page 67, change the radius entry to **1e-3**.
- In Step 3 for the hooks on page 68, click the Chamfer button to create a chamfer instead of a fillet. Type **2e-4** in the **Distance** edit field.

# New Features in the Electromagnetics Module

## *Higher-Order Vector Elements*

---

In addition to linear vector elements, the Electromagnetics Module now makes 2nd and 3rd-order vector elements available. These higher-order vector elements provide increased accuracy in electromagnetic wave simulations. You can select from **Vector - Linear**, **Vector - Quadratic**, and **Vector-Cubic** in the lists of elements for all application modes that use vector elements.

For an example model that uses higher-order vector elements, see “New Model—Inductance of a Power Inductor” on page 28.

## *Vector Elements for All Mesh Element Types*

---

You can now use the linear and higher-order vector elements with all mesh element types. That is, the vector elements work with unstructured meshes as well as mapped, extruded, and revolved meshes.

## *Multigrid Support in Electromagnetic Wave Models*

---

The multigrid solver now supports the use of nested meshes and different element orders for models that include vector elements.

## *Multigrid Support for AV Quasi-Statics and Magnetostatics Models*

---

You can now use the multigrid solver for AV quasi-statics and magnetostatics (vector potential formulation) with explicit gauge fixing. Predefined settings make it easy to take advantage of the accelerated performance that the multigrid solver provides.

## *Transient Quasi-Statics, Vector Potential—Induction Currents*

---

The 3D application mode for induction currents using quasi-statics with a magnetic vector formulation now supports transient analysis in addition to time-harmonic analysis. To access this option from the **New** tab in the **Model Navigator**, select **3D** in the

**Space dimension** list and then in the list of application modes browse to **Electromagnetics Module>Quasi-Statics, Magnetic>Induction Currents>Transient analysis**.

### *Import of SPICE Netlists via COMSOL Script*

With the `spiceimport` script you can import a SPICE netlist into an ODE structure representing the circuit in the file. The script assumes that netlists are in the original Berkeley netlist syntax for devices and models (<http://www.eecs.berkeley.edu/>). Most circuit simulators can export to this format.

Not all device models are implemented, and those implemented do not support the full range of parameters available. It might therefore be necessary to simplify the netlist prior to the import. Currently `spiceimport` supports the devices in Table 1-1 below with the listed limitations.

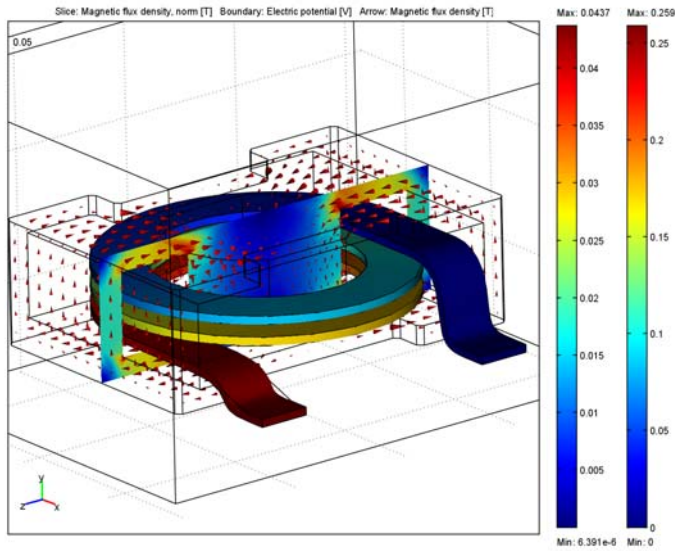
TABLE 1-1: SUMMARY OF SUPPORTED DEVICES

SYMBOL	DESCRIPTION	LIMITATION
R	Resistor	No temperature dependence
C	Capacitor	No voltage and temperature dependence
L	Inductor	No current and temperature dependence
V	Independent voltage source	Support constant sources, pulse sources, and sine sources. Variable names can be used to implement arbitrary expressions by adding them later in Global expressions
I	Independent current source	See above
E	Voltage controlled current source	Gain controlled source
F	Voltage controlled current source	See E device
G	Voltage controlled current source	See E device
H	Voltage controlled current source	See E device
D	Diode	No temperature dependence
Q	Bipolar transistor	Implements parts of the Gummel-Poon transistor model
M	MOS transistor	Implements the MOS transistor model as defined by Shichman and Hodges

## *New Model—Inductance of a Power Inductor*

Power inductors are a central part of many low-frequency power applications. This model shows an inductance calculation on a large 3D geometry using higher-order vector elements and memory-efficient iterative solver settings.

You can download the model and the associated documentation from <http://www.comsol.com/support/updates/>. They are also available in the `updates` folder on the COMSOL 3.2a CD 4.



*Figure 1-6: This plot of a power inductor shows the potential on the coil, the magnitude of the flux density inside the ferrite core, and the direction of the flux as arrows.*

# New Features in the MEMS Module

## *Predefined Multiphysics Couplings for Flow with Species Transport*

---

A suite of predefined multiphysics couplings for flow with species transport is now available in the MEMS Module through its list of application modes in the **Microfluidics** folder. The **Flow with Species Transport** folder includes the following predefined combinations of microfluidic application modes with Convection and Diffusion application modes:

- Non-Isothermal Flow plus Convection and Diffusion
- Incompressible Navier-Stokes plus Convection and Diffusion
- Non-Isothermal Stokes Flow plus Convection and Diffusion
- Stokes Flow plus Convection and Diffusion
- General Laminar Flow plus Convection and Diffusion

All these combinations support transient and steady-state analysis in 2D, axisymmetric 2D, and 3D.

Each predefined coupling provides the velocity components from the microfluidic application mode as the default values for the velocity in the convective part of the Convection and Diffusion application mode.

### **EXAMPLE MODEL—MICROCHANNEL H-CELL**

The Microchannel H-Cell is a new model in the MEMS Module Model Library, that makes use of the predefined multiphysics coupling for flow with species transport. The model treats a microchannel H-cell for separation through diffusion.

## *New Models*

---

The following new models (and documentation) are available at <http://www.comsol.com/support/updates/> and in the `updates` folder on the COMSOL 3.2a CD 4:

### **ELECTROOSMOTIC BIOCHIP**

This example models the electroosmotic flow in a DNA chip. Fluid displacement takes place due to the presence of a charged solution at the walls of the chip's channels, and

the fluid migrates when an electric field is applied. The electroosmotic flow in the model couples the electric field to the Stokes Flow application mode at the boundaries.

#### **THERMOELASTIC DAMPING IN 2D AND 3D**

Thermoelastic damping is an important factor when designing MEMS resonators. The cyclic deformation of the resonator creates local temperature variations, which result in thermal expansion of the material. The expansion appears as damping. These models show how to model thermoelastic damping in a MEMS resonator using fully coupled thermal and structural equations in both 2D and 3D.

#### **MICROCHANNEL H-CELL**

See “Example Model—Microchannel H-Cell” on page 29.

# New Features in the Structural Mechanics Module

## *The Truss Application Modes*

---

Truss application modes in 2D and 3D are now available in the Structural Mechanics Module. You can use trusses to model lattice works, often in combined with beams. This removes the need to use several Euler Beam application modes to model a moment-free connection between beams. Instead you can use beams and trusses. Trusses are available on boundaries in 2D and edges in 3D. A truss is a component only capable of withstanding only axial forces. Using the Truss application mode you can solve hanging cable problems as well as more standard truss problems. The degrees of freedom are the global displacements. The application modes use Lagrange shape functions and tangential derivatives.

The following analysis types are available in the Truss application mode:

- Static analysis
- Nonlinear static analysis
- Frequency response analysis
- Eigenfrequency analysis
- Quasi-static transient analysis
- Parametric analysis
- Transient analysis
- Linear buckling analysis

The Truss application mode also includes these features:

- Coordinate systems on all domain levels
- Large deformations
- Initial strain and stress
- Temperature coupling
- Possibility to constrain an edge to be straight to solve the classic singular problem for trusses.
- Possibility to model hanging cable problems where the edge is not straight

## EXAMPLE MODEL—IN-PLANE TRUSS

The In-Plane Truss example model contains a linear static analysis of a simple 2D pin-jointed truss. The solution to this benchmark problem correlates well with analytical results.

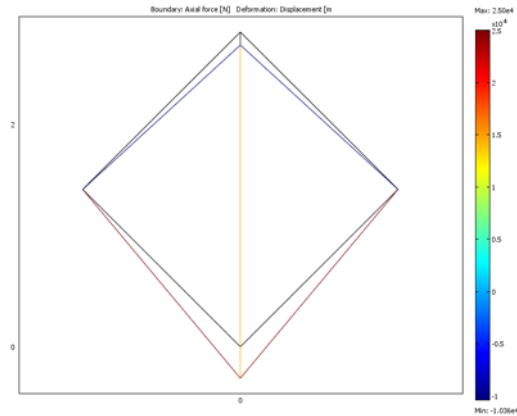


Figure 1-7: Axial forces and deformation for an in-plane truss model.

### *Updated and Corrected Models*

Updated versions of the following models (and the model documentation) are available at <http://www.comsol.com/support/updates/> and in the `updates` folder on the COMSOL 3.2a CD 4.

## ACOUSTIC-STRUCTURE INTERACTION

The eigenvalue formulation in the Acoustics application mode now provides second-order eigenvalues. To match the first-order eigenvalues in the Mindlin Plate application mode, add the following steps before the steps in the section “Subdomain Settings” on page 19 in the *Structural Mechanics Module Model Library*.

### *Subdomain Settings*

- 1 From the **Physics** menu select **Equation System** and then **Subdomain Settings**.
- 2 Copy the expression in the  $\mathbf{e}_a$  edit field and paste it into the  $\mathbf{d}_a$  edit field.
- 3 Type 0 in the  $\mathbf{e}_a$  edit field.
- 4 Click **OK**.

## BLADDER

This updated version of the Bladder model uses realistic values for the Rayleigh damping and individual tolerance settings for improved accuracy. Make the following changes to the step-by-step descriptions in the *Structural Mechanics Module Model Library*:

### Subdomain Settings

Replace Steps 21 and 22 on page 138 with the following steps:

- Click the **Material** tab and select all subdomains to specify the same Rayleigh damping parameters to all subdomains.
- Set  $\alpha_{dM}$  to 1.88 and  $\beta_{dK}$  to 0.016.
- Click **OK**.
- In the **Physics** menu select **Properties**, then **On** in the **Large Deformation** list.

### Computing the Solution

Add the following steps after Step 3 on page 139:

- Type 1e-3 in the **Relative tolerance** edit field.
- Type u 2e-7 v 3e-6 w 5e-7 in the **Absolute tolerance** edit field to specify individual tolerances for each displacement component.

In addition, we recommend that you run the simulation for only a fraction of the full two seconds (0.2 seconds, for example), because of the size of the model.

## COUPLED VIBRATIONS

The model now uses some simplified settings for the density and the normal acceleration.

## TRAFFIC TUNNEL

The updated version of this model shows how to use a Mohr-Coulomb nonlinear material model by making the Drucker-Prager criterion identical to the Mohr-Coulomb criterion. This version also uses more realistic material data and a simplified modeling approach with two application modes instead of three.

## IN-PLANE TRUSS

Example model for the new Truss application mode. See “Example Model—In-Plane Truss” on page 32.

# New Features in COMSOL Script

## *Improved Plots*

---

You can now use logarithmic scales on both the  $x$ -axis and the  $y$ -axis.

## *Support for Formatting and Symbols in Texts*

---

The `text` function can now take formatted strings that include HTML tags, Greek letters, mathematical symbols, and Unicode characters. These formatting options include the strings in plot titles as well as  $x$ -axis,  $y$ -axis, and  $z$ -axis labels (the `title`, `xlabel`, `ylabel`, and `zlabel` functions, respectively).

The `text` function supports the following HTML tags in the text string:

TABLE 1-2: VALID HTML TAGS

HTML TAG	DESCRIPTION
<code>&lt;B&gt; &lt;/B&gt;</code>	Enclosed text is rendered using a bold font.
<code>&lt;BR&gt;</code>	Line break.
<code>&lt;CENTER&gt; &lt;/CENTER&gt;</code>	Centered text.
<code>&lt;I&gt; &lt;/I&gt;</code>	Enclosed text is rendered using an italic font.
<code>&lt;LI&gt;</code>	List item. When the list used is <code>&lt;OL&gt;</code> (an ordered list) the LI element is rendered with a number. When the list used is <code>&lt;UL&gt;</code> (an unordered list) the LI element is rendered with a bullet.
<code>&lt;OL&gt; &lt;/OL&gt;</code>	Ordered list (see also: <code>&lt;LI&gt;</code> ).
<code>&lt;P&gt; &lt;/P&gt;</code>	Paragraph. This tag creates a line break and a space between lines.
<code>&lt;PRE&gt; &lt;/PRE&gt;</code>	Enclosed text preserves spaces and line breaks. The text is rendered using a monospaced font.
<code>&lt;STRIKE&gt; &lt;/STRIKE&gt;</code>	Enclosed text is rendered with a strike-through appearance.
<code>&lt;SUB&gt; &lt;/SUB&gt;</code>	Enclosed text is rendered in subscript with the enclosed text slightly lower than the surrounding text.
<code>&lt;SUP&gt; &lt;/SUP&gt;</code>	Enclosed text is rendered in superscript with the enclosed text slightly higher than the surrounding text.
<code>&lt;TT&gt; &lt;/TT&gt;</code>	Enclosed text is rendered using a monospaced font.

TABLE I-2: VALID HTML TAGS

HTML TAG	DESCRIPTION
<U> </U>	Enclosed text will be underlined.
<UL> </UL>	Unordered list (see also: <LI>).

The text function supports the following Greek character tags in the text strings:

TABLE I-3: VALID GREEK SYMBOL TAGS

TAG	SYMBOL	TAG	SYMBOL
\ALPHA	A	\alpha	$\alpha$
\BETA	B	\beta	$\beta$
\GAMMA	$\Gamma$	\gamma	$\gamma$
\DELTA	$\Delta$	\delta	$\delta$
\EPSILON	E	\epsilon	$\epsilon$
\ZETA	Z	\zeta	$\zeta$
\ETA	H	\eta	$\eta$
\THETA	$\Theta$	\theta	$\theta$
\IOTA	I	\iota	$\iota$
\KAPPA	K	\kappa	$\kappa$
\LAMBDA	$\Lambda$	\lambda	$\lambda$
\MU	M	\mu	$\mu$
\NU	N	\nu	$\nu$
\XI	$\Xi$	\xi	$\xi$
\OMICRON	O	\omicron	$\omicron$
\PI	$\Pi$	\pi	$\pi$
\RHO	P	\rho	$\rho$
\SIGMA	$\Sigma$	\sigma	$\sigma$
\TAU	T	\tau	$\tau$
\UPSILON	Y	\upsilon	$\upsilon$
\PHI	$\Phi$	\phi	$\phi$
\CHI	X	\chi	$\chi$
\PSI	$\Psi$	\psi	$\psi$
\OMEGA	$\Omega$	\omega	$\omega$

The `text` function supports the following math symbol tags in the text string:

TABLE I-4: VALID MATH SYMBOL TAGS

TAG	SYMBOL	TAG	SYMBOL
<code>\approx</code>	≈	<code>\bullet</code>	•
<code>\leq</code>	≤	<code>\partial</code>	∂
<code>\geq</code>	≥	<code>\nabla</code>	∇
<code>\plusminus</code>	±	<code>\sqrt</code>	√
<code>\infty</code>	∞	<code>\integral</code>	∫

In addition to these Greek and math symbols, you can specify additional characters using Unicode numbers. Visit [www.unicode.org](http://www.unicode.org) for more information about Unicode characters.

#### EXAMPLES OF FORMATTED TEXTS

To plot the following mathematical text in the position (1, 2) in the current axes

$$\sin(2\pi x_i) \approx 0,$$

type:

```
text(1,2,'sin(2\pix<SUB>i</SUB>) \approx 0','FontName','Arial',...
'FontSize',16)
```

This example also specifies the font name and size using the optional `FontName` and `FontSize` properties.

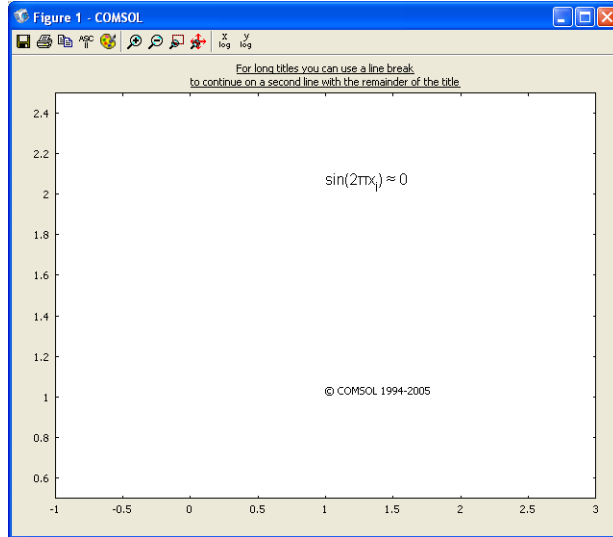
To add a text that includes the copyright symbol, you can use its Unicode:

```
text(1,1,'\u00A9 COMSOL 1994-2005')
```

To add an underlined title to a plot with the text divided into two lines, type:

```
title('<U>For long titles you can use a line break<BR>to continue
on a second line with the remainder of the title</U>')
```

The following figure shows the text strings in this example in a COMSOL Script window.



## *New Functions*

---

The 3.2a release adds a number of new functions to COMSOL Script. For more information about these functions, type `help` followed by the function name at the COMSOL Script command prompt. The following tables list the new functions divided into the following categories:

- Graphics functions (Table 1-5 on page 38)
- Image functions (Table 1-6 on page 38)
- Linear algebra functions (Table 1-7 on page 38)
- Object functions (Table 1-8 on page 39)
- Operating system functions (Table 1-9 on page 39)
- Polynomial and spline functions (Table 1-10 on page 39)
- Sparse matrix functions (Table 1-11 on page 40)
- Specialized mathematical functions (Table 1-12 on page 40)
- Statistics functions (Table 1-13 on page 40)
- Zero-finding function (Table 1-14 on page 40)

## GRAPHICS FUNCTIONS

TABLE I-5: GRAPHICS FUNCTIONS

FUNCTION	DESCRIPTION
bar	Create a bar graph
campos	Control the camera position
camtarget	Control the camera target
camup	Control the camera up vector
camva	Control the camera view angle
clabel	Add labels to a contour plot
contour	Create a contour plot
contour3	Create a 3D contour plot
countourc	Compute a contour data matrix
contours	Compute a contour data matrix
errorbar	Error bar plot
trimesh	Create a mesh plot with triangles
trisurf	Create a surface plot with triangles

## IMAGE FUNCTIONS

TABLE I-6: IMAGE FUNCTIONS

FUNCTION	DESCRIPTION
image	Show an image
imagesc	Show an image using scaled mapping
imread	Read an image from file
imshow	Show an image
imwrite	Write an image to file

## LINEAR ALGEBRA FUNCTIONS

TABLE I-7: LINEAR ALGEBRA FUNCTIONS

FUNCTION	DESCRIPTION
funm	Evaluate a matrix function
logm	Matrix logarithm
ordschur	Reorder Schur factorization

## OBJECT FUNCTIONS

TABLE I-8: OBJECT FUNCTIONS

FUNCTION	DESCRIPTION
clone	Create a copy of an instance of a user-defined class
methods	Get the methods provided by a user-defined class
super	Run superclass constructor
this	Get the instance for which an instance method is run

## OPERATING SYSTEM FUNCTIONS

TABLE I-9: OPERATING-SYSTEM FUNCTIONS

FUNCTION	DESCRIPTION
cputime	CPU time (in seconds)
dir	Get a list of the files in a directory
dos	Run a DOS command
encrypt	Encrypt M-files
isdir	True if a directory exists
mkdir	Create a directory
profile	Generate profiling information
rmdir	Remove a directory
system	Run a system command
unix	Run a system command

## POLYNOMIAL AND SPLINE FUNCTIONS

TABLE I-10: POLYNOMIAL AND SPLINE FUNCTIONS

FUNCTION	DESCRIPTION
delaunay	Delaunay triangulation
delaunay3	3D Delaunay triangulation
mkpp	Make a piecewise polynomial
pchip	Piecewise cubic Hermite interpolation
ppval	Evaluate a piecewise polynomial
spline	Cubic spline interpolation
unmkpp	Extract details from piecewise polynomial

## SPARSE MATRIX FUNCTIONS

TABLE 1-11: SPARSE MATRIX FUNCTIONS

FUNCTION	DESCRIPTION
spdiags	Sparse diagonal or band matrix manipulation
spones	Sparse matrix of ones
sprand	Sparse random matrix with uniformly distributed numbers
sprandn	Sparse random matrix with normally distributed numbers
sprandsym	Symmetric sparse random matrix

## SPECIALIZED MATHEMATICAL FUNCTIONS

TABLE 1-12: SPECIALIZED MATHEMATICAL FUNCTIONS

FUNCTION	DESCRIPTION
betainc	Incomplete beta function
gammainc	Incomplete gamma function

## STATISTICS FUNCTIONS

TABLE 1-13: STATISTICS FUNCTIONS

FUNCTION	DESCRIPTION
corrcoef	Correlation coefficients
cov	Covariance matrix
subspace	Principal angle between subspaces

## ZERO-FINDING FUNCTION

TABLE 1-14: ZERO-FINDING FUNCTION

FUNCTION	DESCRIPTION
fzero	Find a zero of a function

## *Updated and Improved Functions*

- The `text` function supports HTML formatting, Greek letters, mathematical symbols, and Unicode characters. See “Support for Formatting and Symbols in Texts” on page 34 for details.
- The `uint8` function returns an unsigned integer (`uint8` array). All other functions for converting doubles to integers and unsigned integers (`int*`, `uint*`) are unchanged and return doubles.

## *User-Defined Classes*

---

Creating custom classes helps you improve the organization of your code. A class is a custom data type that contains data as well as methods operating on objects of the class. Each class is defined by a `cs1`-file (short for COMSOL Script language), which defines the fields and methods that the class provides. The object model is inspired by the Java class system, but you do not need any previous experience with Java to use it. For complete documentation on user-defined classes and objects in COMSOL Script, see `scriptclasses.pdf`, which is available in the `updates` directory on the COMSOL 3.2a CD 4 and the COMSOL web site at <http://www.comsol.com/support/updates/>.

## *Encrypting M-files*

---

You can protect the contents of custom M-files using the `encrypt` command, which creates encrypted versions of the files in the input arguments. The input files must exist and be valid M-files. For each input file, COMSOL Script creates an MC-file in the current directory (extension `.mc`). When you run an MC-file, it is equivalent to the original M-file, but `encrypt` has scrambled its contents to make it unreadable. Use the optional input argument `-inplace` to make `encrypt` store each MC-file in the same directories as the corresponding M-file.

# Chemical Engineering Module Updates

## *Updated Rising Bubble Model Using Level Sets*

---

A new version of this model includes several improvements for easier modeling and improved accuracy. This version also corrects some errors in the documentation. The following changes are the most important:

- To avoid computing the curvature explicitly in the surface-tension force, you integrate this force by parts (after multiplying with the test functions) using a surface divergence theorem. By doing so you can apply the force by adding the weak contribution

$$\sum_{i=1}^d \sigma \delta(\nabla_s \hat{\mathbf{u}}_i)_i$$

where  $\nabla_s = (\mathbf{I} - \mathbf{nn}^T) \cdot \nabla$  represents the surface gradient operator, and  $\hat{\mathbf{u}}_i$  is the test function for the  $i$ th velocity component. This way of computing the surface-tension force results in better accuracy than the explicit application of the force because it uses only first-order derivatives (of the unknowns).

- The model uses a Heaviside function for a smooth transition of the density at the interface between the fluids. This function now takes a normalized level set function as its first input argument and the scale is a mesh-dependent expression. This makes the Heaviside function work across a wider range and for different meshes.
- The added diffusivity at the phase interface is now also defined using a mesh-dependent expression.

The model file and complete model documentation are available at <http://www.comsol.com/support/updates/> and in the `updates` folder on the COMSOL 3.2a CD 4.

# I N D E X

- 64-bit windows version 2
- A** acoustic-structure interaction model 32
- analyzed geometry
  - creating from mesh 5
- animation
  - with multiple FEM structures 23
- antialiasing 21
- B** Berkeley netlist syntax 27
- beta function 40
- C** Chemical Engineering Module 42
- classes 41
- COMSOL Reaction Engineering Lab 4
- COMSOL Script 34
  - multiphysics scripting support 18
  - new functions in 37
- contour labels 17
- coupling variables
  - frame selection for 6
- CPU time 39
- Create Geometry From Mesh 5
- csl-files 41
- custom fonts 19
- D** deformed mesh
  - minimum quality of 9
- deformed mesh, creating geometry from
  - 5
- Delaunay triangulation 39
- documentation set for COMSOL 3.2a 3
- DOS command 39
- Draw mode, entering 22
- E** Electromagnetics Module 26
- electroosmotic biochip 29
- eliminated stiffness matrix 14
- encrypting M-files 41
- evaluation points
  - selecting 23
- extended mesh
  - information about 13
- F** flow with species transport 29
- frame selection
  - for coupling variables 6
- frames when remeshing 11
- G** gamma function 40
- Gauss points
  - evaluating in 23
- geometry repair 12
- global expression variables 13
- global expressions
  - plotting 15
- graphics functions 38
- Greek characters 35
- greek symbols 35
- H** hanging cable problems 31
- higher-order vector elements 26
- HTML formatting 34
- HTML tags 34
- I** image export 19
- image functions 38
- installing COMSOL 3.2a 2
- L** level sets 42
- line width 21
- linear algebra functions 38
- logarithmic scales 34
- M** math symbols 36
- MEMS Module 29
- mesh frames 12
- mesh optimization 12
- mesh2geom function 6

- minimum quality of deformed mesh 9
  - moving boundaries
    - remeshing 6
  - multigrid solver 26
- N**
  - named single user license, installing 2
  - NASTRAN mesh 14
  - new features
    - in COMSOL 3.2a 5
    - in the Electromagnetics Module 3.2a 26
    - in the MEMS Module 29
    - in the Structural Mechanics Module 31
  - overview 3
  - new functions
    - in COMSOL Script 37
  - new models
    - in the Electromagnetics Module 28
    - in the MEMS Module 29
  - null-space matrix 14
- O**
  - objects 41
  - ODE variables, evaluating 23
  - operating system functions 38, 39
- P**
  - parameterized geometries
    - remeshing 6
  - particle tracing 17
  - plotting global expressions 15
  - polynomial and spline functions 39
  - predefined multiphysics couplings 29
  - profiling information 39
- R**
  - Reaction Engineering Lab 4
  - reference frames 11
  - remeshing
    - for moving boundaries (ALE) 6
- S**
  - scaling of images 19
  - selection of evaluation points 23
  - shallow water equations 24
  - sparse matrix functions 40
  - spatial frames 11
  - specialized mathematical functions 40
  - SPICE netlists 27
  - spline interpolation 39
  - statistics functions 40
  - stop condition 9
  - stop conditions for solvers 15
  - streamline plots 17
  - Structural Mechanics Module 31
  - system commands 39
- T**
  - text symbols 36
  - texts
    - formatting and symbols 34
  - thermoelastic damping 30
  - tick marks 19
  - transient quasi-statics 26
  - trusses 31
- U**
  - undefined operations 15
  - unicode 36
  - updated and corrected models
    - in COMSOL Multiphysics 24
    - in the Structural Mechanics Module 32
  - user-defined classes 41
- V**
  - vector elements 26
- W**
  - wave number 24
  - Windows XP Professional x64 Edition 2
- Z**
  - zero-finding function 40