COMSOL Multiphysics

Release Notes

5.3a

COMSOL

COMSOL Multiphysics Release Notes

Version: COMSOL 5.3a

Contact Information

Visit the Contact COMSOL page at www.comsol.com/contact to submit general inquiries, contact Technical Support, or search for an address and phone number. You can also visit the Worldwide Sales Offices page at www.comsol.com/contact/offices for address and contact information.

If you need to contact Support, an online request form is located at the COMSOL Access page at www.comsol.com/support/case. Other useful links include:

- Support Center: www.comsol.com/support
- Product Download: www.comsol.com/product-download
- Product Updates: www.comsol.com/support/updates
- COMSOL Blog: www.comsol.com/blogs
- Discussion Forum: www.comsol.com/community
- Events: www.comsol.com/events
- COMSOL Video Gallery: www.comsol.com/video
- Support Knowledge Base: www.comsol.com/support/knowledgebase

Part number: CM010001

Contents

Chapter 1: Release Notes

COMSOL Multiphysics 10

General New Functionality 10
New Functionality in the Application Builder 10
New Functionality in COMSOL Server™ 13
New and Improved General Functionality in COMSOL Multiphysics . . . 13
New Functionality in Physics Interfaces 16
New Geometry and Mesh Functionality 16
New and Updated Operators, Functions, and Definitions 17
New Functionality in Studies and Solvers 18
New and Improved Results and Visualization Functionality 21
New Java® API Methods 23
General Backward Compatibility Considerations 23
Backward Compatibility with Version 5.3 24
Backward Compatibility with Version 5.2a 25
Backward Compatibility with Version 5.2 26
Backward Compatibility with Version 5.1 27
Backward Compatibility with Version 5.0 28
Backward Compatibility with Version 4.4 29
Backward Compatibility with Version 4.3b 30
Backward Compatibility with Version 4.3a 31
Backward Compatibility with Version 4.3 33

AC/DC Module 34

New Functionality in Version 5.3a 34
New and Updated Application in Version 5.3a 35
Backward Compatibility with Version 5.2 and older 35
Backward Compatibility with Version 5.0 and older 36

Acoustics Module 38

New Functionality in Version 5.3a 38
New Model Examples in 5.3a 39
Backward Compatibility with Version 5.3 40

Backward Compatibility with Version 5.2. 40
Backward Compatibility with Version 5.1 40
Backward Compatibility with Version 4.3a and Older 41
Backward Compatibility with Version 4.3. 41

Batteries & Fuel Cells Module **42**

New Functionality in Version 5.3a 42
New and Updated Applications and Models in Version 5.3a 43
Backward Compatibility with Version 5.3. 43
Backward Compatibility with Previous Versions 44

CAD Import Module, Design Module, and LiveLink™ Products for CAD **45**

New Functionality in Version 5.3a 45
New Functionality in LiveLink™ for AutoCAD® in Version 5.3a 45
New Functionality in LiveLink™ for Inventor® in Version 5.3a 45
New Functionality in LiveLink™ for Revit® in Version 5.3a 45
LiveLink™ for Solid Edge® 45
LiveLink™ for SOLIDWORKS® 46
Backward Compatibility with Previous Versions 46

CFD Module **47**

New Functionality in Version 5.3a 47
New Model in Version 5.3a 48
Backward Compatibility with Version 5.0. 48
Backward Compatibility with Version 4.4. 48
Backward Compatibility with Version 4.3b 49
Backward Compatibility with Version 4.3a 50
Backward Compatibility with Version 4.3. 50

Chemical Reaction Engineering Module **55**

New Functionality in Version 5.3a 55
New Models in Version 5.3a 55
Backward Compatibility with Version 5.0 and Earlier 56

Corrosion Module **60**

New Functionality in Version 5.3a 60
New and Updated Models in Version 5.3a 60

Backward Compatibility with Version 5.3 61
Backward Compatibility with Earlier Versions 61

ECAD Import Module **62**
New Functionality in Version 5.3a 62
Backward Compatibility with Version 5.2a 62
Backward Compatibility with Version 5.0 62

Electrochemistry Module **63**
New Functionality in Version 5.3a 63
Backward Compatibility with Version 5.3 63
Backward Compatibility with Previous Versions 63

Electrodeposition Module **64**
New Functionality in Version 5.3a 64
New and Updated Models in Version 5.3a 64
Backward Compatibility with Version 5.3 64
Backward Compatibility with Previous Versions 65

Fatigue Module **66**
Backward Compatibility with Version 5.3 66

Geomechanics Module **67**
New Functionality in Version 5.3a 67

Heat Transfer Module **68**
New and Improved Functionality in Version 5.3a 68
New Applications in Version 5.3a. 72
Updated Applications in Version 5.3a 73
Backward Compatibility with Version 5.3 74
Backward Compatibility with Version 5.2a 75
Backward Compatibility with Version 5.2 75
Backward Compatibility with Version 5.1 76
Backward Compatibility with Version 5.0 77
Backward Compatibility with Version 4.4 79
Backward Compatibility with Version 4.3b 81
Backward Compatibility with Version 4.3a 81
Backward Compatibility with Version 4.3 82

Obsolete Features 84

MEMS Module **86**

New Functionality in Version 5.3a 86

Backward Compatibility with Version 5.2. 86

Backward Compatibility with Version 4.3b 87

Microfluidics Module **88**

Backward Compatibility with Version 4.3a 88

Backward Compatibility with Version 4.3 88

Mixer Module **89**

New Functionality in Version 5.3a 89

Molecular Flow Module **90**

Backward Compatibility with Version 5.0 and Earlier 90

Multibody Dynamics Module **91**

New Functionality in Version 5.3a 91

New Applications in Version 5.3a. 91

Nonlinear Structural Materials Module **94**

New Functionality in Version 5.3a 94

New Applications in Version 5.3a. 94

Optimization Module **95**

Backward Compatibility with Version 4.3a 95

Particle Tracing Module **96**

New Functionality in Version 5.3a 96

New Applications in Version 5.3a. 97

Backward Compatibility with Version 5.3 and Earlier 97

Backward Compatibility with Version 5.2a and Earlier 98

Backward Compatibility with Version 5.2. 98

Backward Compatibility with Version 5.1 99

Backward Compatibility with Version 5.0 and Earlier 100

Pipe Flow Module **101**
Backward Compatibility with Version 5.1 and older 101

Plasma Module **102**
New Functionality in Version 5.3a 102
New and Updated Applications and Models in Version 5.3a 102
Backward Compatibility with Version 5.3 and Earlier 102
Backward Compatibility with Version 5.2 and Earlier 103
Backward Compatibility with Version 5.1 104
Backward Compatibility with Version 4.4 104

Ray Optics Module **105**
New Functionality in Version 5.3a 105
New Applications in Version 5.3a. 106
Backward Compatibility with Version 5.3 and Earlier 106
Backward Compatibility with Version 5.2a and Earlier 107
Backward Compatibility with Version 5.2 108
Backward Compatibility with Version 5.0 109

RF Module **111**
New Functionality in Version 5.3a 111
New and Updated Models and Applications in Version 5.3a 112
Backward Compatibility with Version 4.3a 112

Rotordynamics Module **114**
New Functionality in Version 5.3a 114
New Applications in Version 5.3a. 115

Semiconductor Module **116**
New Functionality in Version 5.3a 116
New Models in Version 5.3a 117

Structural Mechanics Module **118**
New Functionality in Version 5.3a 118
Backward Compatibility with Version 5.2 122
Backward Compatibility with Version 4.3b 123
Backward Compatibility with Version 4.3a 123

Subsurface Flow Module

Subsurface Flow Module 124

New Functionality in Version 5.3a 124

New Models in Version 5.3a 125

Wave Optics Module 126

New Functionality in Version 5.3a 126

New Model in Version 5.3a 127

Backward Compatibility with Version 5.2 and Earlier 127

Material Library 129

New and Updated Material Data in Version 5.3a 129

LiveLink™ for MATLAB® 130

New Functionality in Version 5.3a 130

Backward Compatibility with Version 5.2. 131

Backward Compatibility with Version 5.0. 131

LiveLink™ for Excel® 132

New Functionality in Version 5.3a 132

The COMSOL API for Use with Java® 133

COMSOL 5.3a API Changes 133

COMSOL 5.3 API Changes 133

COMSOL 5.2a API Changes 134

COMSOL 5.1 API Changes 135

COMSOL 5.0 API Changes 137

COMSOL 4.4 API Changes 139

COMSOL 4.3b API Changes 140

COMSOL 4.3a API Changes 141

COMSOL 4.3 API Changes 141

COMSOL 4.2a API Changes 142

COMSOL 4.1 API Changes 143

COMSOL 4.0a Java API Changes 144

1

Release Notes

COMSOL Multiphysics® version 5.3a includes extended and improved versions of the Application Builder and COMSOL Server™, as well as updates and improvements for all COMSOL Multiphysics® add-on products. These *Release Notes* provide information regarding new functionality in version 5.3a for all COMSOL®software products.

COMSOL Multiphysics

General New Functionality

- The security preference **Allow external processes and libraries** has been split into **Allow external processes**, **Allow external libraries**, and **Allow external MATLAB® functions**.
- A new Cluster Setup Validation app helps you check that the cluster connections for COMSOL Multiphysics® and COMSOL Server® work.

New Functionality in the Application Builder

IMPROVEMENTS TO THE FORM EDITOR

- The form editor drag-and-drop functionality has been updated to support copy and abort. Copy is enabled by holding down the Ctrl key, and dragging can be aborted with the Esc key.
- Toggle toolbar items have been added to toolbars, tables, and graphics form objects. Those form objects also show tooltips and texts of toolbar items separately in the table. The Toolbar Items dialog box, used to add toolbar items to Table form objects, shows the tooltip if the text does not exist for toolbar items.
- The Editor Tools input option for array and matrix properties now adds a Table form object.
- When running applications with a ribbon in the cross-platform user interface (or in a web browser) in previous versions of COMSOL Multiphysics, the software has rendered this as a row of toolbar buttons without text. Now, the user interface displays these buttons in a more logical and visually appealing manner.
- You can copy form objects by dragging them and then pressing the Ctrl key.
- Big buttons can now leave out text and still look good. The image is centered rather than at the very top of the button as in previous versions.
- An orientation setting has been added for Radio Button form objects.
- The grid mode in the form editor now includes limited support for dragging a box to select multiple cells. It works when you start dragging either in an empty cell or outside the grid. For the sketch mode, holding Shift while dragging to select now adds to the current selection.
- The *On data change* event from a Slider form object can be suppressed while the thumb is being dragged.

- Button and Toggle button form objects now include a manual height option.
- Toolbar form objects can now have buttons with both an image and an icon.
- The toolbars in form objects can now be given a size: small or large.
- Most settings in a physics interface can now be controlled with Model Data Access.
- The Edit Argument dialog box now uses a table to define a matrix argument that has an undefined column count. The table has buttons for adding rows and columns, which simplify the process of defining initial values for 2D array declarations with an undefined column count and method input argument default values.

IMPROVEMENTS TO THE METHOD EDITOR AND APPLICATION LANGUAGE

See also the *Application Programming Guide* for a complete overview of the application language for programming methods in the Application Builder.

- When creating physics interfaces in the method editor, a code completion request (Ctrl+Space) now lists all available interfaces with brief descriptions and links to the documentation.
- The **Inputs** table in a method's **Settings** window now has a **Description** column that can be used to annotate method parameters. Code suggestion for Application Builder methods now displays method parameters and the tooltip includes the information entered in the **Description** column.
- The method `clearModel(model)` no longer deletes method calls.
- A new utility method, `getComsolVersion()`, which can be used in application and model methods, is now available. It returns the current COMSOL® software version as a string.
- A `createDirectory(`*`directorypath`*`)` utility method, which can be used to create directories, is now available.
- You can now provide end users of an application with better error messages when the cause of the error came internally from COMSOL Multiphysics using

  ```
  error("An error occurred", cause);
  ```

- There are now two ways to clear the graphics window in an application: `useGraphics(null, "/form1/graphics1")` and `graphicsWindow.set("source", (ModelEntity) null)`

  These methods revert the graphics window to the state it would have if it had no source when launching the application, which is a 3D empty Graphics window.

- The message utility methods have been improved with method overloads for all scalar and array declaration types.
- The try/catch language template has been replaced by the following code:

```
try {
  // code that might fail with exception

} catch (Exception e) {
  // handle possible exception
}
```

- Three new utility methods for matrices: sort, replaceRow, and replaceColumn. The sort method orders the columns by their row values from top to bottom.
- Boolean expressions such as 1<2 can now be used in command sequences when setting Boolean properties and declarations.
- Column headers of Table form objects can be changed at runtime through methods using the headers(string array) and showheaders(boolean) properties.
- A **Description** column is now available in the table of input arguments for methods. The description appears for code completion and in the **Settings** window for a corresponding method call in the Model Builder.

The following methods and improvements have been added for use with model methods:

- A static method, selectNode, has been added to the API. This method can be used from model methods to select a node in the Model Builder. The method takes a model entity as an argument.
- The following methods controlling the main COMSOL® software progress are now accessible from model methods:
  - setProgress(int)
  - setProgress(int, String)
  - setProgressInterval(String, int, int)
  - resetProgress()
- The method clearModel(model) no longer deletes method calls.
- The Application Builder debug log can now be viewed in the Model Builder.

New Functionality in COMSOL Server™

- Automatic login for COMSOL Server™ is now available. You can place the responsibility for authenticating users external to COMSOL Server™ by activating the **Log in users automatically without requiring passwords** check box.
- You can now export and import COMSOL Server™ preferences. The exported preferences will automatically be migrated to the current version on import.
- Administrators and power users now have the ability to send notifications to users.
- The title, description, and thumbnail of an application can now be overridden in COMSOL Server™ via the Edit Application dialog, available from the Application Library.
- Administrators can now test and save new login configurations using a new administrative tool, **Login configuration** under the **Administration** side menu.
- Connecting to COMSOL Server™ from a MATLAB®client has been made easier. There is now an installer that installs the required files on the client computer. For Windows®, this is included in the COMSOL Client installer. For macOS and Linux®, there is a zip file called `COMSOL53a_client_api.zip` that contains the required files.

New and Improved General Functionality in COMSOL Multiphysics

- Copy-paste of model components and physics interfaces is now possible within a COMSOL Multiphysics model and also between COMSOL Multiphysics sessions. You can also insert components and physics interfaces from other models.
- For mouse navigation in the COMSOL Desktop® on Windows®, you can now use a 3D SpaceMouse® from 3Dconnexion®. All 3D mouse types in http://www.3dconnexion.eu/ are supported.
- Recovery files now appear in a **Recovery Files** window in the COMSOL Desktop®, and that window appears directly when starting COMSOL Multiphysics®, if any recovery files exist.
- Improved and extended support for reduced-order modeling and model reduction, with a new **Model Reduction** node in studies; better support for model reduction in the modal and AWE solvers; and creation of **Reduced Model** nodes under **Global Definitions**, which can be exported and imported. Also, various system matrices can be exported using the **System Matrix** node under **Derived Values**. A new tutorial model, *Thermal Controller, Reduced Order Model*, is available to demonstrate how to use reduced-order modeling.

- In all text fields (and table cells) that previously supported the use of global parameters, you can now use user-defined functions (including functions defined by materials, thermodynamic property packages, M-files in MATLAB®, and external DLLs). In particular, you can use user-defined functions in the **Parameters** nodes (under **Global Definitions** and **Results**) and in the **Local Parameters** node under a geometry part. User-defined functions and material functions with one argument also appear as functions to apply in the **Range** dialog box. Also, global parameters can be used in the **Parameters** node under **Results**.
- The rotation center functionality for rotating 3D geometries has been improved. The rotation center is now automatic by default, but you can control it manually and switch between the two states by pressing the R key when the Graphics window has focus. You can also constrain rotation using the mouse by pressing and holding the X, Y, and Z keys to lock the *x*-, *y*-, or *z*-axis, respectively. Pressing those keys can be done while rotating. You can also lock the rotation center using the new **Lock rotation center** check box in the 3D **View** nodes. From the **Graphics and Plot Windows** page in the **Preferences** dialog box, you can turn some of this functionality on and off, and you can make the rotation center icon always show or show only on mouse actions.
- A new zoom-to-point feature is available for quickly zooming into details in 3D models. When you Ctrl+middle-click, you now zoom into the point that the mouse is pointing to in the Graphics window.
- The plus (+) and minus (-) keys can now be used to move the selection to the next or previous entity that you can reach. The up arrow and down arrow keys, which provided the same functionality in previous versions, can also be used but are deprecated and will not be available for this functionality in future versions.
- You can now add moving mesh features to any COMSOL Multiphysics model. They are available from the **Moving Mesh** submenu when right-clicking the **Definitions** node under a **Component** and from the **Definitions** toolbar. The moving mesh features control the spatial frame and therefore potentially apply to all physics in the model. They can be used to study both stationary states and time-dependent deformations where the geometry changes its shape due to the dynamics of the problem.
- You can now run model methods (and methods that include inputs) using **Method Call** nodes that you add under **Global Definitions**. Method calls are also available for batch jobs and other job sequences, and it is possible to run a method call using a `-methodcall` argument to the `comsolbatch` command.
- In Windows®, the COMSOL Desktop® uses a new and improved tree component, which makes it possible to change the label directly in the tree by pressing F2 when

a node is selected in the tree. Also, when dragging and dropping nodes, the tree scrolls and expands items automatically.

- For the COMSOL Desktop® in Windows®, the following improvements are included in version 5.3a. However, the retained user interface settings can be lost if the cache becomes full.
  - Table column resizing has been improved. It is now possible to resize table columns without making the Settings window wider first.
  - When the user changes the height of user interface controls, the new size is retained while the model is open. The width of table columns is also retained.
  - Changing the height of a user interface control is now supported for list boxes and tree controls. Resizing is enabled on a case-by-case basis depending on the specific Settings window.
  - The pop-up table (which shows up when clicking **>>** in the top-left corner of many tables) retains its width and height.
  - The position of the window's vertical scrollbar is also retained.
- An option to show context help directly in an external web browser is now available as a preference setting.
- Preview plots of user-defined functions now appear in a separate **Function Plot** window.
- The **Parameters** node under **Global Definitions**, for defining global model parameters, is now always present.
- If you save a model that was previously saved in an earlier version of COMSOL Multiphysics, you now get a question about if you want to continue to save the model file, thereby overwriting the original file with a new file that has been converted so that it can only be opened in version 5.3a. You can turn off this question, if desired, by clearing the **Warn before overwriting a file saved by an older version of COMSOL** check box on the **Files** page in the **Preferences** dialog box.
- The **Selection List** window now displays the number of selected entities above the list of entities (domains or boundaries, for example).
- Icons for user library folders in the Application Libraries and Part Libraries windows can now be customized: A PNG-file named `folder.png` placed in a user library folder will override the default icon. The image should be 16-by-16 pixels.

- It is now possible to clear the thumbnail image used in a model file.
- You can now specify `-portfile` *<path>* when launching a COMSOL Multiphysics server using `comsolmphserver` to have COMSOL Multiphysics write its server port to the given *<path>*.

New Functionality in Physics Interfaces

- Physics features with an empty selection that are not added by default now get a warning decoration and a tooltip that informs you that the selection is empty.
- The PDE, Boundary Elements interface has been reimplemented and improved. It now includes an a coefficient for Helmholtz-type equations. New postprocessing variables have been added for up and down fields on boundaries. Postprocessing using kernel-evaluated fields has been improved with smoothing near boundaries. Backward compatibility is provided for MPH-files but not for the Java API.
- In the Wave Form PDE interface, a new Interior Flux Split boundary flux condition is available. It makes it possible to specify the numerical flux separately on the upside and downside of interior boundaries.
- It is now possible to add Global Equations features in the Events interface.
- Many physics interfaces in COMSOL Multiphysics include stabilization: fluid flow interfaces, Transport of Diluted Species, and Heat Transfer, to mention a few. The stabilization in most of these interfaces has been updated for time-dependent problems. The update improves the stability and accuracy for small time steps. Additional dependent variables, typically named `dt2Inv` (sometimes with a suffix), have been introduced to measure intrinsic time scales within the physics.

New Geometry and Mesh Functionality

GEOMETRY FUNCTIONALITY

- Improved **Interpolation Curve** feature: It is now possible to specify tangent directions (a clamped spline) at the ends of the curve. In the case when the relative tolerance is 0, the algorithm has been changed, which can give a somewhat different shape of the curve.
- In the **Remove Details** node, there is a new **Thin Domains** check box, selected by default, under **Details to Remove**. If you select this check box, the operation will remove domains with a thickness less than the specified detail size.

- For the **Partition Domains** feature, when using geometry objects to partition the domains, a new **Keep objects** check box is available to control if the objects used to partition with should be kept in the resulting geometry.
- For the **Partition Objects** feature, new **Keep objects to partition** and **Keep tool objects** check boxes are available to control if the objects used to partition with and the tool objects should be kept in the resulting geometry. The **Keep objects to partition** check box replaces the **Keep input objects** check box in earlier versions.
- **Copy Selection** and **Paste Selection** buttons are now be available for all selections in the geometry sequence.
- The **Insert Expression** button is now available in geometry **Part** node and in the **Local Parameters** subnode.
- The **Mesh Parts** and **Geometry Parts** nodes are now created automatically when adding a mesh part or geometry part.

MESHING FUNCTIONALITY

- The ability to evaluate based on an initial expression is now available in the **Size Expression** node. It lets you adapt the mesh size to an expression that can be computed from the initial values of your problem, using any variable the physics, materials, and so on may define. A typical application is to adapt the mesh size to a (nonconstant) wavelength.
- When exporting and importing meshes to a COMSOL Multiphysics file (`.mphbin` or `.mphtxt`), you can now include mesh selections. Mesh selections that you import can be renamed and used in, for example, postprocessing.
- In the **Copy** mesh node's settings, if you have parameterized the geometry or mesh and plan to run a parametric sweep, you can now select the **Build source mesh automatically** check box to ensure that the source mesh and geometry are updated before copying.
- The reversed TrafficLight color table is now the default color table for mesh plots.
- For STL mesh import, a repair tolerance is now available. The repair tolerance assures that the imported mesh contains no holes or triangles with a side that is shorter that the tolerance.

New and Updated Operators, Functions, and Definitions

- New nodes for defining matrices and matrix operations — matrix inverse, matrix diagonalization, and matrix decomposition (SVD) — are available under **Definitions>Variable Utilities** in a model component. The matrices can be used as inputs to the solvers, and variables for the matrices make it possible to use them during postprocessing using a matrix evaluation, for example.

- A new **Participation Factors** feature is available under **Definitions>Variable Utilities** in a model component for computing participation factors in structural simulations. This improved functionality for computing participation factors replaces the earlier support for participation factors in the eigenvalue solver.
- New units:
  - `Ah` (for ampere-hour or 3600 C) and `Wh` (for watt-hour or 3600 J). You can use SI prefixes. For example, `3.5[kWh]` is 3500 Wh.
  - `mmH2O` (millimeter of water) as a pressure unit equal to 9.80665 Pa.
- For a Ramp function, you can now use different smoothing zones at the ramp's location and cutoff.
- A new `gpeval` operator, a built-in operator for Gauss point evaluation.

New Functionality in Studies and Solvers

STUDIES AND STUDY STEP FUNCTIONALITY

- Study nodes can be now be moved to rearrange their order.
- Units are now included when saving and loading parameters in a **Parametric Sweep** node.

SOLVER FUNCTIONALITY

- Improved performance for several solvers:
  - The MUMPS and PARDISO direct solvers now have an option for reusing the preordering.
  - The algebraic multigrid (AMG) solver now has an option for reusing prolongators.
  - The Vanka, SCGS, and SOR line solvers now have an option for reusing data. This option stores data that would otherwise be recomputed in each nonlinear iteration. Additionally, SOR line now has an option for reusing lines.

    For all these performance improvements, there is a slightly higher memory peak.
- An improved **Combine Solutions** node now provides options for combining a solution using weighted summation and to remove solutions based on excluding or including solutions using an explicit list or an implicit logical expression. For these methods, you can also clear the source solutions.

- The **Parametric** solver now includes a **Parameter value run order** list with a new default option: **Automatic**. When you use it, the parametric solver performs an analysis of the cost of changing the values of involved parameters, and the run order of the parameter values is modified when it is found to be more efficient.
- You can now distribute the parametric solver by sending one parameter value to each compute node, as well as for the Eigenvalue, Eigenfrequency, Mode Analysis, and Linear Buckling study steps.
- New options for controlling when a solver sequence should be generated during a parametric sweep are available in the Settings window for the **Parametric Sweep** node. From the **Default solver sequence generation** list, you can choose to use global parameters or use the first, last, or each parameter tuple.
- For the linear iterative solvers, the error handling mechanism has been improved to contribute to a general robustness of the computation. For example:
    - Elimination of a possible premature termination.
    - Obtaining extra accuracy for the linear system of equations for the sake of fewer nonlinear iterations. A new residual tolerance setting is available in the solver. The **Iterative** solver node is valid for the left preconditioning of the GMRES, BiCGStab, and conjugate gradients solvers.
    - Avoiding warnings and errors related to ill-conditioned preconditioners.
- The previous solution solver for a time-dependent problem now includes a **Damping factor** check box and an input field allowing a damping value between 0 and 1 (default 1). By default, damping is not active.
- For solving stationary problems using a Newton solver, the **Fully Coupled** node now includes a more stringent termination criterion: **Solution and residual**.
- The strict time stepping algorithm has been improved so that it avoids unnecessarily small time steps.
- The in-core memory required by the PARDISO solver for out-of-core computations is now increased automatically.
- A new parameter, maximum column size factor, has been added for the SAI preconditioner. It limits the maximum number of nonzero elements of each column in the SAI preconditioner matrix. It can improve convergence for models with many nonlocal couplings, which can occur in BEM models.

- The sensitivity solvers have been improved and can now compute the correct sensitivities when the objective depends on accurate boundary fluxes.
- The **Modify physics tree and variables for study step** check box in the study step nodes' Settings windows under **Physics and Variables Selection** has been renamed to **Modify model configuration for study step** to better reflect that it also provides control of various features under **Component>Definitions**; for example, for configuring the model for a specific simulation.

CLUSTER AND BATCH FUNCTIONALITY

- **Method Call** nodes can now be added under Batch, Parametric Sweep, and Sequence nodes in job configurations. This way, you can run model methods (without user interface commands) from **Method Call** nodes under **Global Definitions** in such jobs.
- Cluster computing has been improved with the following enhancements:
    - A new option that writes the command line to a file instead of starting it. This file can then be used to run the command on another computer.
    - The **File transfer command** list has a new option: **File list**. This option stores a list of all files that should be transferred to another computer where the batch job runs.
    - A new option, the **Remote hosts list**, which selects if the remote host list should be read from the remote hosts table or from a file. The remote hosts file points to a file on the server side (in client-server; otherwise, on the current computer) that contains the list of hosts to run on remotely.
    - A new command-line switch on Linux® `-mpihosts` takes a comma-separated list of hostnames.
- New COMSOL Multiphysics batch commands: `-stop` and `-cancel` for stopping and canceling batch jobs, and `-clearmesh` and `-clearsol` for clearing all meshes and solutions, respectively.
- New COMSOL Multiphysics batch command: `-mode`. When it is `batch` (the default), all Batch and Cluster Computing settings are ignored. When `-mode` is `desktop`, Batch and Cluster Computing settings will be used.
- A new batch option `-jobfile` allows a text file with a list of input and output filenames. Each pair of input and output filenames will be run consecutively as a batch job.
- In addition, the list can be updated continuously during the execution of a batch process.

- PBS (Portable Batch System) is now available as a scheduler type in the Cluster Computing and Cluster Sweep features.
- For cluster computing, WCCS 2003 is no longer supported.

New and Improved Results and Visualization Functionality

NEW AND IMPROVED GRAPHICS AND PLOT FUNCTIONALITY

- Plot data can now be saved in the model, which can be useful for plots that take a long time to recompute but do not contain a large amount of data (such as some 1D plots). In the main Results node, you can specify if the plot data should be saved in the model automatically when it is useful, always, never (similar to the behavior in earlier versions of COMSOL Multiphysics), or manually for each plot group.
- There are now **Plot First** and **Plot Last** buttons on the toolbar at the top of the plots' and plot groups' Settings windows for quickly moving to the first or last time, parameter value, or eigenmode.
- In the main **Results** node's Settings window, there is now an **Automatic regeneration of plots when the solver completes** check box, which you can select to solve a study and compute the data and its dependent plots in one step.
- Plots of max/min markers are now also available at points as Max/Min Point plots. In addition, you can specify a background color, add a frame, and add a unit for the maximum and minimum values.
- In graph plots, the cycle of line style, color, or marker can be reset so that it is possible to, for example, synchronize the line colors when comparing two sets of curves.
- You can also customize the colors used in the color cycle for graph plots. If there is a file `graphcyclecolors.txt` in the preferences directory, then it is used. Otherwise, `$DISTR/data/colors/graphcyclecolors.txt` is used.
- It is now possible to generate images using the graphics card. Such hardware-accelerated image export runs much faster and can make it possible to export larger images, but it can become unstable if the image size exceeds or is close to the allowed limit (for example, 4000-by-4000 pixels). To enable hardware-accelerated image export, open the **Preferences** dialog box, and, on the **Graphics and Plot Windows** page under **Image export**, clear the **Use software rendering for image export** check box.
- Filter subnodes can now be added to 1D plots.
- Arrow plots are now available at points.

- Principal stress and principal strain plots are now available on lines (edges) in 2D and 3D. Note that if you add a Principal Stress Line plot to an old model, then you do not get any default values from the physics. Update the solution or recompute the study to get the default values.
- A new Impulse Response 1D plot type is available with the Acoustics Module.
- A new *Cividis* color table uses yellow and blue colors in a color table that is suited for red-green colorblindness, for example.
- It is now possible to convert an added headlight to a directional light.
- 2D Histogram plots now include Range settings for controlling the color range and the data range.
- Ellipses can now be used as the point type in Point Trajectories plots.
- Solution-dependent expressions (instead of only global parameters) can now be used for the far-field evaluation radius in Far Field plots.
- Legends are now active by default in probe plot windows, and you can turn the legends on and off in the probe plot window while solving.
- The automatic legends in Point Graph and Line Graph plots can now be configured.
- The placement of arrows and markers in, for example, 3D arrow and principal stress/strain plots can now be in the Gauss points.
- Principal stress/strain plots where the placement was in the mesh nodes in earlier versions now place the stress/strain markers in the mesh element centers to get better plots without jumps or averaging due to several elements sharing the same node.
- Solution-dependent expressions can now be used in streamline start positions.
- You can now specify a potentially solution-dependent expression for the azimuthal mode number in Revolution 2D and Sector data sets.
- In Histogram plots, there is now a choice between volume-based and element-based histograms.
- It is now possible to choose between several sets of stress/strain data in the Principal Stress/Strain plots.
- You can now sweep over phases when evaluating Whirl plots.
- The default smoothing method in plots is now to smooth inside material domains.
- In directivity plots, you can now use a linear frequency scale.

IMPROVED DATA EXPORT AND ANIMATIONS

- WebM, a movie file format for high-quality animations, is now available.
- The **Compute volume integral** and **Compute surface integral** check boxes for computing derived values in axisymmetric models is now active by default for new models.
- Solution-dependent expressions (instead of only integers) can now be used for the regular grid settings in data export features.

NEW AND IMPROVED DATA SETS

- New Receiver 2D and Receiver 3D data sets for computing receiver data in ray acoustics, used for impulse response plots. These data sets are available with the Acoustics Module.
- Solution-dependent expressions (instead of only integers) can now be used for the regular grid settings in cut point data sets.
- An **Extra Time Steps** section has been added to the Settings window for the Intersection Point data sets for adding and rendering extra time steps.
- You can now choose between symmetric and antisymmetric vector transformations in the Mirror data sets.

IMPROVED REPORT GENERATION

- The Report Generator output for physics interfaces, physics features, and multiphysics couplings now more closely reflects the Settings window layout in the COMSOL Desktop®. In particular, the show/hide settings in the **Show** toolbar menu for **Discretization**, **Stabilization**, and **Advanced Physics Options** are respected.

New Java® API Methods

- The `ModelUtil.modelsUsedByOtherClients()` method returns the tags of the models used by other clients.
- The `view.geom()` and `view.getSDim()` methods return the geometry and the space dimension for the view, respectively.

General Backward Compatibility Considerations

COMSOL Multiphysics version 5.3a can open MPH files saved from COMSOL Multiphysics versions 4.0–5.3.

COMSOL Multiphysics version 5.3a can run Java files saved from COMSOL Multiphysics versions 4.0–5.3. However, the Java files may need to be modified in accordance with information following in this document and may need to be recompiled with the `comsol compile` command in version 5.3a.

Backward Compatibility with Version 5.3

PDE, BOUNDARY ELEMENTS INTERFACE

The PDE, Boundary Elements interface has been reimplemented and improved. Backward compatibility is provided for MPH files but not for the Java API.

STABILIZATION IN SOME PHYSICS INTERFACES

Model files created in version 5.3 and earlier retain the old stabilization formulation. The new formulation is obtained by replacing the physics interface with a new one. The improved stabilization breaks Model Java-file backward compatibility for time-dependent models that contain one of the affected physics interfaces. Please contact COMSOL Support to learn how to retain backward compatibility for your specific Model Java-file.

STORE SOLUTION ON DISK

The **Store solution on disk** option has been removed.

FUNCTIONS IN GEOMETRY FEATURES

The button **Rebuild with Updated Functions** has been removed from the following geometry features: **Parametric Curve**, **Parametric Surface**, and **Sweep**. Instead, use the **Refresh** or **Clear Functions** button in the function features.

In the API, the method `importData()` is obsolete for the geometry features `ParametricCurve`, `ParametricSurface`, and `Sweep`. Instead, use the `refresh()` method in the function features.

PARTICIPATION FACTOR

Support for participation factors in the eigenvalue solver has been replaced by a new and improved **Participation Factors** feature, which is available under **Definitions>Variable Utilities** in a model component for computing participation factors in structural simulations.

INTERPOLATION CURVES

For interpolation curves, in the case when relative tolerance is zero (which is the default), the algorithm in 5.3a has changed so that the shape of the curve becomes somewhat different (while still interpolating the given points).

ITERATIVE SOLVERS

For the iterative solvers, the error handling mechanism has been improved to contribute to a general robustness of the computation. The improved mechanism is enabled per default and can give the following effects for nonlinear iterations:

- Increased total number of linear iterations
- Failure with the error "Divergence of the linear iterations", if the preconditioner is extremely ill-conditioned.

UNITS IN THE SIZE EXPRESSION NODE

The size expression is now interpreted in the model component's unit system rather than in the geometry's unit system. Also, the spatial coordinates (x, y, z, and so on) are now always in the component's unit system. In version 5.3, when evaluating on a grid, the spatial coordinates were in the geometry's unit system.

STRESS/STRAIN PLOTS

Principal stress/strain plots now evaluate the stresses and strains in the centers of mesh elements instead of in the mesh nodes.

Backward Compatibility with Version 5.2a

FRAME DEFINITIONS

In version 5.3a, all frames are always defined. If you open models from version 5.2a or earlier, there will be a **Permanently Define All Frames** button under **Frames** in the General section of the **Component** nodes. When you click the button, all frames are defined, and the button disappears.

FRAME CONTROL

In previous versions of COMSOL Multiphysics, it was possible to apply multiple frame-controlling physics on the same selection. For example, it was possible to add two Solid Mechanics interfaces (with the **Include geometric nonlinearity** option enabled) on the same selection and solve. In this case, an "override rule" was applied between the controlling physics, effectively meaning that the last physics in the tree had control of the frame deformation for the overlapping domains. This situation could cause unexpected results.

In version 5.3, multiple physics interfaces controlling the same frame are not allowed on the same selection and will cause an error when trying to solve. However, it is still possible to use multiple frame-controlling physics with overlapping selection, but you now have to explicitly disable frame control on all but one of the physics. This gives you better control on what is going on. The functionality to disable frame control for physics has been incorporated in the **Settings** window for study steps to allow choosing different physics to control the frame in different study steps.

Frame-Scoped Variables

Certain variables pertaining to frame deformation are now defined with "frame prefix". For example, the relative element volume is now called `spatial.relVol`. Previously, these variables were added by the physics interface controlling the frame deformation so they used a "physics prefix" (for example, `ale.relVol`). For backward compatibility of old models, the physics interfaces still define the old variables, but they are aliases of the new frame variables.

API SYNTAX FOR ERROR HANDLING

The old syntax for an error

```
feature("foo").feature("prob1").feature("error1")
```

is no longer supported. Instead, use the following documented syntax:

```
feature("foo").problem("error1")
```

FREE QUAD MESHING

Models saved in version 5.2a with the free quad tessellation method set to **Automatic** (in a **Free Quad** feature) or face meshing method set to **Quadrilateral** (in a **Swept** feature), will, when opened in 5.3, have the tessellation method or face meshing method set to **Legacy version 5.2a** or **Quadrilateral (legacy version 5.2a)**, respectively.

Backward Compatibility with Version 5.2

FREE QUAD MESH

The new free quad meshing algorithm is used in new models, but for models created in earlier versions, the legacy algorithm is used instead. The default for the new `method` property is therefore `auto` in new models, and `legacy52` in migrated models.

GEOMINFO CHECK() METHOD

The return value of the `check()` method in GeomInfo has been removed. Now, `check()` throws an error if the geometry is invalid.

THE CONST PROPERTY IN SOLVERS

The `const` property available for many solvers has been removed in version 5.2a and replaced with the `cname` and `clist` properties, which are string arrays for the constant names and corresponding constant values, respectively. For the **Dependent Variables** node, the new property `initparametersmethod`, which can be set to `"auto"` (the default) or `"manual"`, has been added to control automatic synchronization of values of parameters to use for initial expressions. The **Dependent Variables** node now also includes the `cname` and `clist` properties.

Backward Compatibility with Version 5.1

CREATING SELECTIONS

In version 5.2, the `selresult` property replaced the `createselection` property. `createselection` is still supported for backward compatibility.

SELECTIONS IN PART INSTANCES

For backward compatibility for selections in part instances, there is a **Keep noncontributing selections** check box, cleared by default, in the **Selection Settings** section of the Settings window for **Part Instance** nodes. If you select the **Keep noncontributing selections** check box, the **Keep column** is disabled, and the selection is kept if the **Contribute to** value is **None**. In the COMSOL API, the default is an active Keep noncontributing selections setting.

MESH PARTS

For backward compatibility regarding STL/VRML import, and for COMSOL API compatibility, the old user interface for mesh import (with a specified file name and import properties) and the corresponding properties are available in the COMSOL API but not in the COMSOL Desktop, unless it is the active option. This means that the old user interface can only be reached if the filename is set in combination with the STL import type (the API type `stlvrml`), which cannot be done from the new user interface.

DORMAND-PRINCE 5 TIME-STEPPING SOLVER

The Dormand-Prince 5 Runge-Kutta solver in 5.1 does not use field norm scaling in version 5.1 when estimating errors. This means that the errors of, for instance, an ODE might become very small when solved together with a field with many DOFs, and the time steps taken might then be too large. In version 5.2a, the Runge-Kutta solver uses field norm scaling, which means that old models using the Dormand-Prince 5 might need tighter tolerances to produce results in earlier versions that are similar to those in 5.2a.

Backward Compatibility with Version 5.0

MESH IMPORT

The method used to automatically partition the boundary of imported meshes in 3D has been improved. If you have an existing model, you can work with it without being affected. However, if you click the **Import** button of the **Import** feature in the meshing sequence, the file is read again and the new partitioning method is used. The exception is if the **Import** feature had **Boundary partitioning** set to **Manual**. In this case, the modified parameter values are preserved under the **Feature detection** setting, which uses the same algorithm as in version 5.0.

COMSOL tries to map the old selection on boundaries to the new boundaries, but it is not always possible to do accurately when new faces have appeared or old faces have disappeared. You may have to manually review and update boundary, edge, and point selections after reimporting the mesh.

If you have a Java® or MATLAB® program that imports meshes, the number of geometric entities may have changed compared to older versions.

SECURITY SETTINGS

In version 5.2a, the **Allow external process and libraries** check box on the **Security** page in the **Preferences** dialog box is cleared by default to not allow applications to start external processes on the computer. The default setting in 5.0 is set to allow such external processes.

DISPLAY OF MAX/MIN MARKERS IN PLOTS

The display of max/min markers in plots is now off by default to make plots fit better when using a small graphics canvas. In previous versions, the display of max/min markers in plots was on by default.

MERGED MPH-FILE FORMAT

The MPHAPP file and MPH file formats have been merged since version 5.1, and all application files use the `.mph` file suffix. You can still open MPHAPP files created in version 5.0.

Backward Compatibility with Version 4.4

COMSOL SERVER

COMSOL Server from version 4.4 is now called the COMSOL Multiphysics Server.

MESH IMPORT

An edge in an imported mesh that has a common start and end vertex, or that lacks start and end vertices, is now split into two edges with distinct start and end vertices. For meshes where this happens, the numbering of all geometric entities may change when the mesh is rebuilt.

For MPH files created in earlier versions, selections are automatically updated with the new entity numbers.

LOCAL COORDINATE VALUES

For model components created in version 5.0 and onward, the variable names `xi1`, `xi2`, and `xi3` are reserved for the predefined local coordinate variables. To access the previous behavior (that local coordinate variables do not exist), you can use the API method `model.modelNode(`*`<tag>`*`).defineLocalCoord(false)`.

TRANSPORT OF DILUTED SPECIES

Running Java®-files from previous versions may fail due to the new default name for the Transport of Diluted Species interface. This can be avoided by adding a command that specifies the identifier in accordance with the name of the interface. For example, when creating an interface using the old name (`chds`):

```
model.physics().create("chds", "DilutedSpecies", "geom1",
                       new String[][]{{"c"}});
```

Add the following line to specify the identifier accordingly:

```
model.physics("chds").identifier("chds");
```

The Java® API syntax for creating and accessing vectors and tensors in the Transport of Diluted Species interface has changed as well as the syntax for setting physics properties. This does not affect MPH files. See the backward compatibility notes for the Chemical Reaction Engineering Module for additional information that also applies to the Transport of Diluted Species interface.

DOCUMENTATION

The *COMSOL Multiphysics Programming Reference Manual* replaces the *COMSOL API for Use with Java® Reference Manual.*

Backward Compatibility with Version 4.3b

MODEL NODES ARE NOW COMPONENT NODES

The **Model** nodes (as they were called in previous versions), which contained separate model components in a model file, are called **Component** nodes in version 5.2a.

CHANGES TO PARAMETRIC SWEEPS

Old models that use stationary parametric sweeps are loaded with the **Reuse solution for previous step** list set to **Yes**. The **Run continuation for** list is set to the parameter used, unless the continuation algorithm would not have been used for this model in previous versions (for example, if multiple parameters are used or if the parameter list is not monotonous).

CHANGES TO THE PHYSICS SELECTION IN STUDY SETTINGS

The names of the states of a physics interface in the physics tree, which you can modify under **Physics and Variables Selection** in the study steps' settings windows, have changed:

- **Provide Degrees of Freedom** is now called **Disable in Solvers**.
- **Disable** is now called **Disable in Model**.

REVISED FORMULATION FOR LAMINAR INFLOW AND LAMINAR OUTFLOW

The formulations of the laminar inflow and laminar outflow conditions have been corrected. The modified formulation gives a more accurate mass flux. Some models may now produce a slightly different flow field.

Laminar inflow and laminar outflow are available in the fluid flow physics interfaces in the following modules:

- Batteries & Fuel Cells Module
- CFD Module
- Corrosion Module

- Electrochemistry Module
- Electrodeposition Module
- Heat Transfer Module
- Microfluidics Module
- Plasma Module
- Subsurface Flow Module

NEW DEFAULT FOR GRAPHICS OPTIMIZATION

Under **Graphics and Plot Windows** in the **Preferences** dialog box, the default settings in the **Optimize for** list is now **Quality** instead of **Performance**.

VELOCITY/ACCELERATION INTEGRATION VARIABLE

For the Solid Mechanics interface (and all related multiphysics interfaces) and the Truss interface, the help variable `u0` (velocity integration variable) is used in **Prescribed Velocity** and **Prescribed Acceleration** features for Time Dependent study types. This variable computes the displacement for each point where the condition is prescribed. It is changed to use the full feature scope in order to avoid collisions in cases where several such features exist within the same model component. As a result, when opening and running an old model that uses such features together with a segregated solver, an error message appears, stating that not all dependent variables occur in at least one of the segregated solver steps. The relevant action is to manually add the velocity integration variable to the segregated step containing the corresponding displacement field. Alternatively, you can regenerate any affected solver sequence.

MATH LIBRARIES ON AMD PROCESSORS

MKL is now the default math library on AMD processors. Switching to the ACML math library for AMD processors might improve performance in some cases.

Backward Compatibility with Version 4.3a

ERROR ESTIMATION CHANGES IN THE SOLVERS

The **Automatic** method for **Check error estimate** for direct linear solvers and for **Validate error estimate** has changed. For nonlinear and time-dependent problems, the underlying nonlinear solver does not accept termination for a linear solution step that does not fulfill the error estimate (unless the step size is very small). The motivation for this change is that the old method can lead to premature termination of the nonlinear solution process, which in some cases introduces a large error for the computed solution.

This change can cause the solvers to take more nonlinear steps for stationary problems and more time steps for time-dependent problems, and it may also lead to convergence problems. For such cases, use the **No** method to obtain the old behavior. However, doing so can hide numerical problems and potentially lead to large numerical errors.

TERMINATION CRITERION FOR STATIONARY SOLVERS

The default termination criterion for stationary solvers has changed. In 4.3a, the settings corresponded to **Solution**; now, the default is **Solution or residual**. This change in default termination criterion might affect models created in earlier versions of COMSOL Multiphysics if you regenerate the solver sequence in 5.2a.

BACKWARD EULER INITIALIZATION TIME STEP

A new setting in the **Advanced** section of the settings window for the **Time-Dependent Solver**, called **Fraction of initial step for Backward Euler**, provides an option for entering a dimensionless quantity that determines the size of the time step for the backward Euler method (in terms of the initial step). This value can improve the accuracy of the initialization step but can also affect the start-up of some models. The default value is 0.001 (this differs from earlier versions, which used a value of 1). When opening models created in version 4.3a or earlier, the value for this fraction is set to 1 to maintain compatibility with those versions.

LOADING EXTERNAL PHYSICS BUILDER JAR FILES

External physics builder JAR archives compiled with earlier versions of COMSOL Multiphysics include a manifest file that contains a reference to the CDO library, which has been removed. To load such JAR files in version 5.2a, you must first do one of the following:

- Delete the META-INF/MANIFEST.MF file in the archive source on the file system, and then recompile the JAR archive.
- Manually remove the line with `org.eclipse.emf.cdo` in the META-INF/MANIFEST.MF file in the JAR archive. You can do this directly in a file archive manager such as 7-Zip or similar.

HIGHLIGHTING GEOMETRY OBJECTS WHEN DRAWING IN 2D

When you have drawn one geometry object on top of another object, toggling of the highlighting of these objects occurs when you click several times. It is important that you click without moving the cursor in a position where the objects overlap in order to toggle. For example, if you draw a circle (C1) and then draw another smaller circle (C2) inside of C1, then first clicking on C2 may highlight C1. If you click for a second time without moving the cursor, it will highlight C2. In previous versions, you would move the cursor after the first click in order to highlight C2.

Backward Compatibility with Version 4.3

NEW TERMINOLOGY FOR CONSTRAINT TYPES

The following constraint types have new names in version 5.2a:

- *Bidirectional, symmetric* is now *Apply reaction terms on: All physics (symmetric)*.
- *Unidirectional* is now *Apply reaction terms on: Individual dependent variables.*

WEAK CONSTRAINTS UPDATE FOR THE LAMINAR FLOW INTERFACE

The weak constraint formulations for the following boundary conditions have been updated:

- Symmetry
- The Slip option in the Wall feature

These boundary conditions are now formulated using the same set of Lagrange multipliers as all of the other boundary conditions in the Laminar Flow interface. The Lagrange multiplier `un_lm` has been removed.

Models saved in version 4.3 will include `un_lm` until the model is re-solved. In some cases, occurrences of `un_lm` in the solver sequence must be replaced manually. This is the case if `un_lm` was the only Lagrange multiplier component in a segregated group or the only Lagrange multiplier component of a Vanka smoother. Alternatively, you can generate a new automatic solver sequence. Models saved in versions earlier than 4.3 must either be re-solved in version 5.2a for postprocessing, or opened and re-saved in version 4.3 before being opened in version 5.2a.

AUTOMATICALLY CREATED IDENTITY/CONTACT PAIRS

The pairs created by the Form Assembly feature in the geometry can differ from their forms in 4.3 for certain 3D and 2D geometries. The pairs should now be the same for the CAD and COMSOL representations.

AC/DC Module

New Functionality in Version 5.3a

MAGNETIC FIELDS, NO CURRENTS, BOUNDARY ELEMENTS INTERFACE

The new Magnetic Fields, No Currents, Boundary Elements interface is based on the boundary element method (BEM). It solves for the scalar magnetic potential and can be used as a standalone physics interface for the modeling of permanent magnets with linear, constant, and homogeneous properties. It also provides multiphysics coupling features, allowing for combined finite element (FEM) and BEM modeling of more complex scenarios when used together with the Magnetic Fields, No Currents interface or the Magnetic Fields interface.

ELECTROSTATICS, BOUNDARY ELEMENTS INTERFACE

The Electrostatics, Boundary Elements interface has been updated and improved. Support has been added for electrostatic force calculations, and there are new postprocessing variables (up/down fields) on boundaries. Postprocessing using kernel-evaluated fields has been improved with smoothing near boundaries. Backward compatibility is provided for MPH files but not for the Java API.

ADDITIONAL NEW FUNCTIONALITY

The Magnetic Fields, No Currents and Rotating Machinery, Magnetic interfaces now support the Frequency Domain study.

The Magnetic Fields, No Currents interface has a new feature, Magnetic Scalar Potential Discontinuity, that can be used for introducing edge current loops in the otherwise current-free formulation. This feature is available when enabling Advanced Physics Options.

A new material model for the modeling of soft permanent magnets has been added to the Magnetic Fields; Magnetic Fields, No Currents; and Rotating Machinery, Magnetic interfaces. A generic example material, Demagnetizable Nonlinear Permanent Magnet with the approximate properties of AlNiCo 5, has been added to the AC/DC material database to serve as a template for user-defined materials supporting the new material model.

The new Surface Magnetic Current Density boundary condition has been added to the Magnetic Fields interface and specifies a surface magnetic current density at both exterior and interior boundaries. Magnetic current density is described by a 3D vector. However, because it flows along a surface, it can be alternatively represented for more efficient modeling. To achieve this, the current density is projected onto a boundary surface and neglects its normal component. The new boundary condition has been provided for special modeling situations, such as for modeling electric dipoles.

New and Updated Application in Version 5.3a

ROTATING MACHINERY 3D TUTORIAL

The *Rotating Machinery 3D Tutorial* example (ACDC_Module/Motors_and_Actuators/rotating_machinery_3d_tutorial) has been updated, showing how to model a laminated rotor introducing insulating boundary conditions in the Rotating Machinery, Magnetic interface.

TUTORIAL EXAMPLES ON THE COMPUTATION OF MAGNETIC FORCES

Two new tutorial examples on the computation of magnetic forces have been added to the AC/DC Application Library (the Electromagnetic Force Verification Series): *Force Calculation 2 — Magnetic Force BEM FEM* and *Force Calculation 3 — Magnetic Torque BEM FEM*. This series of tutorial models will soon be extended with a *Force Calculation 1— Introduction* model, available for download using the Application Library Update. Using the Magnetic Fields, No Currents and Magnetic Fields, No Currents, Boundary Elements interfaces, the boundary element method (BEM) and the finite element method (FEM) are compared to analytical models in a mesh convergence study. For several mesh sizes, the quality of the computed Maxwell surface stress tensor is investigated. The total force and torque are determined. The models serve as an introduction to the boundary element method for magnetostatics.

Backward Compatibility with Version 5.2 and older

COIL FEATURES

Multi-Turn Coil features in old models will be mapped to the new **Coil** feature using the Homogenized Multi-Turn Conductor model. **Single-Turn Coil** features in old models will import as is with a warning that the feature is obsolete and will be removed in future versions.

The **User Defined Coil Geometry** subnode, which is available for 3D **Coil** nodes with the User-Defined Coil type, is now a domain feature. You can now select coil input and output boundaries using the new **Input** and **Output** subnodes. The selection is set up correctly when opening old models.

REMOVED THE OLD INFINITE ELEMENTS FEATURES

The old **Infinite Elements** feature, which are obsolete since version 4.2, have now been discontinued and will be automatically removed when opening old models.

Backward Compatibility with Version 5.0 and older

COIL CURRENT CALCULATION

When models saved in previous versions are opened in version 5.1, the Coil Current Calculation study steps are migrated to Coil Geometry Analysis. Old solutions stored in the model can still be postprocessed, and the solver sequence will be regenerated automatically (with a Stationary solver) the first time the study is solved.

The Coil Geometry Analysis study step (formerly Coil Current Calculation) will now solve for all of the coils in the active interfaces. To solve only for specific coils (specified using the `CoilName` property), set the `SpecifyCoil` property to `1`.

The default solver sequence generated by the Coil Geometry Analysis is different in version 5.1. Code that accesses specific solver features in the generated solver sequence may need to be reviewed.

OTHER COIL IMPROVEMENTS

Harmonic Perturbation subnodes under coil features (**Single-Turn Coil**, **Multi-Turn Coil**) are now global features, so the call to the create method should use the appropriate space dimension (`-1`):

```
model.physics("mf").feature("stcd1").create("hp1",
   "CoilHarmonicPerturbation", -1);
```

Global features have no selections, so code that accesses the selection of the Harmonic Perturbation features may need to be reviewed.

Some of the improvements in the 3D **Multi-Turn Coil** features may require a review of existing code that uses the COMSOL API.

Subnodes required to set up the coil features are now added automatically. Existing code that uses the coil features may need to be updated.

The parameters `eCoil` and `length` have been moved from the **Multi-Turn Coil** features (boundary and domain) to the new subfeature `UserDefinedCoilGeometry`.

OTHER IMPROVEMENTS

New functionality introduced in version 5.1 is disabled by default when opening models created in previous versions:

- New boundary conditions for **Gauge Fixing** features
- Accurate coil voltage calculation

This functionality can be enabled using the appropriate inputs in the Settings window. Refer to the documentation for the individual features for more details.

Acoustics Module

New Functionality in Version 5.3a

The following new functionality is available:

PRESSURE ACOUSTICS, BOUNDARY ELEMENTS INTERFACE

The boundary element method (BEM) is introduced to the Acoustics Module with the new Pressure Acoustics, Boundary Elements interface. The interface is suited for frequency-domain simulations solving the Helmholtz equation with constant-valued material properties. The interface is available in 2D and 3D and is implemented in a scattered-field formulation with the option to add a background pressure field to model scattering problems. The interface is fully multiphysics enabled and can be coupled seamlessly with the physics interfaces that are based on the finite element method (FEM). This includes coupling to vibrating structures, with the Acoustic-Structure Boundary multiphysics coupling, and to FEM acoustic domains, with the new Acoustic BEM-FEM Boundary multiphysics coupling. This approach allows modeling, in a FEM-BEM framework, using the strength of each formulation adequately.

IMPULSE RESPONSE AND RECEIVER DATA SET

It is now possible to postprocess the impulse response from a Ray Acoustics simulation. This is done by combining the new Receiver data set, which collects the ray information, and the new Impulse Response plot, which reconstructs and visualizes the impulse response based on the receiver data.

PRESSURE ACOUSTICS, TIME EXPLICIT INTERFACE

The Pressure Acoustics, Time Explicit interface is a new interface based on the discontinuous Galerkin or dG-FEM formulation. It is solved using a time-explicit method that is very memory efficient and has a low memory consumption. The interface is used to solve large transient linear acoustic problems containing many wavelengths. It is suited for time-dependent simulations with arbitrary time-dependent sources and fields. The interface includes a Background Acoustic Field option for modeling scattering problems. Absorbing layers are used to set up effective nonreflecting-like boundary conditions. The exterior scattered far field can be calculated by combining the Far-Field Calculation feature with a Time to Frequency FFT study step. The interface exists in 2D, 2D axisymmetric, and 3D.

IMPROVED STABILIZATION FOR LINEARIZED EULER

New and improved numerical stabilization methods have been added to the Linearized Euler interfaces. The new default stabilization scheme is the Galerkin least squares (GLS) stabilization, which highly improves the stability and convergence for solutions with coarse meshes.

ABSORBING LAYERS FOR LINEARIZED EULER IN THE TIME DOMAIN

Absorbing layers are now available to be used with the Linearized Euler, Transient interface. This makes it simpler to set up open boundaries in the time domain. The absorbing layers use a combination of three techniques to set up effective nonreflecting boundary conditions.

PLANE WAVE DECOMPOSITION

There is now a built-in option to solve plane wave scattering problems in 2D axisymmetric models using a Plane Wave Expansion approach. This option automatically expands the applied plane wave, in a Background Pressure Field or an Incident Pressure Field feature, into its cylindrical harmonics in terms of the circumferential mode number.

PLANE WAVE (MONOCHROMATIC)

The Pressure Acoustics, Transient and the Pressure Acoustics, Time Explicit interfaces now have a built-in Plane wave (monochromatic) option to define monochromatic plane waves as a background pressure field or an incident pressure field.

IMPROVED SOLVER SUGGESTIONS

In this release, the automatically generated solver suggestions have been improved for several multiphysics applications involving acoustics interfaces. The Transient Solver Settings section, available in all transient acoustic interfaces, has been improved and made more intuitive. Moreover, when a transient acoustics interface is involved in a multiphysics model, the transient solver settings defined for the acoustics interface are now automatically used when solving the coupled problem.

New Model Examples in 5.3a

BESSEL PANEL

The Bessel panel is a way to arrange a number of loudspeakers so that the angular sound distribution resembles that of a single speaker. This model combines five Bessel panels in the same pattern to approximate a purely radial sound field. The speakers are driven with different signals, some of them in counter-phase. This results in an approximately homogeneous polar far-field distribution. The model uses a BEM-FEM approach to solve the radiation from the idealized speaker panel.

LUMPED LOUDSPEAKER DRIVER USING LUMPED MECHANICAL SYSTEM

This is a model of a moving-coil loudspeaker where a lumped parameter analogy represents the behavior of the electrical and mechanical speaker components. The Thiele-Small parameters (small-signal parameters) serve as input to the lumped model. In this model, the mechanical speaker components such as moving mass, suspension compliance, and suspension mechanical losses are modeled using the Lumped Mechanical System interface.

Backward Compatibility with Version 5.3

For the Linearized Euler, Transient interface, initial conditions are only used for the dependent variables. The initial conditions for the time derivatives have therefore been removed. For a model with fields rho, u, and p, remove the Java® lines setting drho/dt, du/dt, and dp/dt (these parameters change names when the field names change).

Backward Compatibility with Version 5.2

The default settings have changed for the Incident Pressure Field in the Pressure Acoustics interfaces. Add the following line to obtain the old behavior of this feature when `ipf1` is a subfeature to pwr1:

```
model.physics("acpr").feature("pwr1").feature("ipf1").
set("c", "acpr.c_c");
```

Backward Compatibility with Version 5.1

- Plane Wave Radiation has been removed from the Linearized Potential Flow, Frequency Domain and Linearized Potential Flow, Transient interfaces in 1D axisymmetry.
- A new Lagrange multiplier variable (`ta.lm_slip`) has been added to the slip conditions in the Thermoacoustics, Frequency Domain interface. If you run a model with a segregated solver, you will need to add this variable to the group containing the velocity degree of freedom. You can also regenerate the default solver. In the API, you need to add `comp1_ta_lm_slip` to the declaration of the segregated step; for example, adding:

```
model.sol("sol1").feature("s1").feature("se1").feature("ss1").
set("segvar", new String[]{"comp1_p", "comp1_u", "comp1_T",
"comp1_ta_lm_slip"});
```

- The Waveguide end impedance option has been removed from 2D, 1D axisymmetric, and 1D as it had no physical meaning in these space dimensions.

Backward Compatibility with Version 4.3a and Older

- The old **Perfectly Matched Layer** (PML) node that is located under the physics node has been discontinued. From the next version on, only the **Perfectly Matched Layer** nodes defined under the **Definitions** node are supported.
- In order for the old **Perfectly Matched Layer** (PML) node to be backward compatible, the PML feature must be placed after any domain **Monopole Source**, domain **Dipole Source**, or **Background Pressure Field** nodes.

Backward Compatibility with Version 4.3

- The symbols for volume sources for the acoustics monopole (Q_m; was Q) and dipole ($\mathbf{q}_d$; was $\mathbf{q}$) volume sources for pressure acoustics have changed.
- The default value for the **Typical wave speed** property in the Acoustic-Piezoelectric Interaction, Frequency Domain interface has changed to 343 m/s.

Batteries & Fuel Cells Module

New Functionality in Version 5.3a

The following new functionality is available:

- Improved concentration model inputs handling in the Lithium-Ion Battery, Battery with Binary Electrolyte, and Single Particle Battery interfaces when using the Materials node to define the model parameters. The concentration input is now generally matched by default to the electrolyte salt concentration for the electrolyte parameters, to the particle concentration for the particle intercalation parameters, and to the particle surface concentration for the porous electrode reaction parameters.
- Heat of mixing can now be added to the heat sources calculated by the Particle Intercalation node (Lithium-Ion Battery and Battery with Binary Electrolyte interfaces) and Positive/Negative Electrode nodes (Single Particle Battery interface). In the Batteries & Fuel Cells Application Library, the *Internal Short Circuit of a Lithium-Ion Battery* and *1D Lithium-Ion Battery Drive-Cycle Monitoring* models, and all lithium-ion battery models in the Thermal Management branch have been updated to include heat of mixing.
- Improved diffusion model implementations in the Transport of Concentrated Species interface. Using the Mixture-averaged and Fick's Law diffusion models, the molecular flux can now be formulated in terms of the mole fraction gradient or the mass fraction gradient. A diffusion model correction has also been implemented. This correction ensures that the net diffusion, with respect to the mass averaged velocity, is zero.
- A new Baker-Verbrugge diffusion option has been added to the Particle Intercalation node (Lithium-Ion Battery and Battery with Binary Electrolyte interfaces).
- A new Lithium insertion kinetics option has been added in the Electrode Reaction node in the Lithium-Ion Battery interface.
- The Initial Cell Charge Distribution now accounts for Additional Porous Electrode Material nodes when setting the initial battery state-of-charge, or cell voltage, in a Current Distribution Initialization study step (Lithium-Ion Battery and Battery with Binary Electrolyte interfaces).

- A new Electrochemical Heating Multiphysics node, which automatically couples both boundary and domain heat sources and temperature between an Electrochemistry branch interface and a Heat Transfer interface.
- A new Electrode Current boundary node has been added to the Electrode Shell interface.
- The Reactions node used on Porous Electrode domains in the Tertiary Current Distribution, Nernst-Planck interfaces now adds source terms for the electrode potential.
- New units: Wh and Ah.
- Dissolving-depositing species concentrations are no longer solved for by default for stationary studies.

New and Updated Applications and Models in Version 5.3a

- A new *Lithium Battery Designer* app is available in the Batteries & Fuel Cells Application Library.
- A new *Heterogeneous Lithium-Ion Battery* model has been added to the Batteries & Fuel Cells Application Library.
- A new model example, *Parameter Estimation of a Time-Dependent Lumped Battery Model*, has been added to the Batteries & Fuel Cells Application Library (requires the Optimization Module).
- The *Internal Short Circuit of a Lithium-Ion Battery* and *Ohmic Losses and Temperature Distribution in a Passive PEM Fuel Cell* tutorials in the Batteries & Fuel Cells Application Library have been updated to use the new Electrochemical Heating Multiphysics node.
- The *Lithium-Ion Battery with Multiple Intercalating Electrode Materials* model in the Batteries & Fuel Cells Application Library has been updated to use the Baker-Verbrugge diffusion model and Initial Cell Charge Distribution node.

Backward Compatibility with Version 5.3

The default tags of the Secondary and Tertiary Current Distribution interfaces have been changed to `cd` (old tag was `siec`) and `tcd` (old tag was `tcdee`), respectively. When running the Java API, an explicit command to set the tag of the interface to the old default tag may have to be added to get Java API files created prior to version 5.3a to run. For example:

```
model.component("comp1").physics("siec").tag("siec");
```

TRANSPORT OF CONCENTRATED SPECIES INTERFACE

See Chemical Reaction Engineering Module.

SURFACE REACTIONS INTERFACE

See Chemical Reaction Engineering Module.

CAD Import Module, Design Module, and LiveLink™ Products for CAD

New Functionality in Version 5.3a

The CAD file import functionality included with these products has been extended to support new versions for some of the supported file formats (see under Read from File, CAD on www.comsol.com/products/specifications/cad).

CAD IMPORT MODULE GEOMETRY KERNEL UPGRADE

The CAD Import Module, the Design Module, and the LiveLink™ products for CAD utilize the Parasolid® geometry kernel from Siemens PLM for solid modeling operations, geometry repair, and defeaturing. (Without these products, a COMSOL-native geometry modeling kernel is used.) The CAD Import Module released with COMSOL 5.3a includes an upgraded version of the Parasolid kernel. As a result, a number of stability issues have been fixed, which makes the import of CAD models and solid operations more robust.

New Functionality in LiveLink™ for AutoCAD® in Version 5.3a

The LiveLink™ interface now supports AutoCAD® 2018.

New Functionality in LiveLink™ for Inventor® in Version 5.3a

The LiveLink™ interface now supports Inventor® 2018.

New Functionality in LiveLink™ for Revit® in Version 5.3a

The LiveLink™ interface now supports Revit® 2018.

LiveLink™ for Solid Edge®

NEW FUNCTIONALITY IN VERSION 5.3A

The LiveLink™ interface now supports Solid Edge® ST10.

BACKWARD COMPATIBILITY WITH VERSION 5.0

The default value for the `keepfree` property of the `LiveLinkSolidEdge` function is now set to `on`. Previously, the default was set to `off`.

LiveLink™ for SOLIDWORKS®

NEW FUNCTIONALITY IN VERSION 5.3A

The LiveLink™ interface now supports SOLIDWORKS® 2018.

BACKWARD COMPATIBILITY WITH VERSION 5.0

The default value for the `keepfree` property of the `LiveLinkSOLIDWORKS` function is now set to `on`. Previously, the default was set to `off`.

Backward Compatibility with Previous Versions

BACKWARD COMPATIBILITY WITH VERSION 5.2

Geometry Repair Tolerances

The geometry repair tolerance can be automatic, relative, or absolute in version 5.2a. In previous versions, the repair tolerance was always a relative tolerance.

- If you have not set the relative repair tolerance `repairtol`, you will get the new default repair tolerance type: `auto`. This will result in a valid geometry in almost all cases. In extreme cases, the result can have a different topology than in 5.2.
- In rare cases, if you have set the `repairtol` tolerance, version 5.2a can fail when building the geometry. If so, set `repairtoltype` to `auto` (select **Automatic** from the **Repair tolerance** list) to get the 5.2 behavior.

CFD Module

New Functionality in Version 5.3a

NEW REALIZABLE K-EPSILON INTERFACES

A new Realizable k-ε turbulence model has been implemented for all fluid flow interfaces that support RANS models (except the High Mach Number Flow, Euler-Euler and Rotating Machinery, Fluid Flow interfaces).

IMPROVED BUBBLY FLOW AND MIXTURE MODEL INTERFACES

The Bubbly Flow and Mixture Model interfaces now support interior walls. They also support all turbulence models that Single Phase Flow supports and they support automatic wall treatment for all turbulence models except k-ε and Realizable k-ε.

MORE FLEXIBLE ROTATING MACHINERY, FLUID FLOW INTERFACE

The functionality previously provided by separate Rotating Machinery, Fluid Flow interfaces is now embedded in the Single Phase Flow interfaces. The Rotating Machinery, Fluid Flow entries in the Model Wizard now add a Single Phase Flow interface and a Rotating Domain feature under Definitions, Moving Mesh in the Model Builder tree. This new design offers a more flexible modeling approach as well as new functionality for the modeling of rotating machinery equipment for fluid flow applications.

FULLY DEVELOPED FLOW OPTION

A new Fully Developed Flow option has been added to the Inlet boundary condition feature in the Single Phase Flow interface. This new boundary condition is similar to the previously provided Laminar Inflow condition, but in addition to laminar flow, it can also be used in conjunction with the following turbulence models:

- Algebraic yPlus
- L-VEL
- k-ε
- Realizable k-ε
- Low Re k-ε
- Spalart-Allmaras

BUOYANCY-INDUCED TURBULENCE

Buoyancy-induced turbulence can now be added to all *k*-based RANS models. It is activated in the Gravity feature.

UPDATED STABILIZATION

All fluid flow physics interfaces (except High Mach Number Flow and Euler-Euler) as well as species and heat transport have updated stabilization for time-dependent models. The updated stabilization provides increased accuracy and robustness for small time steps. Multiphysics model files from previous releases open with the old stabilization. The new stabilization can be accessed by replacing the fluid or transport physics interface with a new one.

New Model in Version 5.3a

A *Flow in a Hydrocyclone* tutorial model that demonstrates the capability of the v2-f turbulence model has been added.

Backward Compatibility with Version 5.0

EULER-EULER MODEL, LAMINAR FLOW

The results from models using the Euler-Euler Model, Laminar Flow interface may differ in version 5.2a due to the more general formulation for the viscous stress in the momentum equation for the dispersed phase. The behavior in the old formulation may be reproduced by dividing the dispersed phase viscosity, μ_D, by the variable `ee.phidPos`.

TRANSPORT OF CONCENTRATED SPECIES

See Chemical Reaction Engineering Module.

Backward Compatibility with Version 4.4

INLET FEATURE

The old inlet features still exist, but have been excluded from the physics context menu. Old models retain old **Inlet** features, but adding a new **Inlet** feature comes with the new functionality. Old Model Java® files create **Inlet** features of the old types. The Normal stress condition can still be prescribed on inlets using an **Open Boundary** or a **Boundary Stress** feature.

NEW NAMES FOR MASS TRANSPORT INTERFACES

Running Model Java® files from previous versions may fail due to the new default names in the Transport of Diluted Species and Transport of Concentrated Species interfaces. This can be avoided by adding a command that specifies the identifier in accordance with the Name of the interface. For example, when creating a Transport of Concentrated Species interface using the old name (`chcs`):

```
model.physics().create("chcs", "ConcentratedSpecies", "geom1",
                       new String[][]{{"wO2", "wH2O", "wN2"}});
```

Add the following line to specify the identifier accordingly:

```
model.physics("chcs").identifier("chcs");
```

WALL DISTANCE EQUATION

The old Continuity node is still included in the interface but is excluded from the feature list. Hence, old models retain their old Continuity node when opened in version 5.2a and old Model Java®files are fully backward compatible.

EULER-EULER MODEL, LAMINAR FLOW

The results from models using the Euler-Euler Model, Laminar Flow interface may differ in version 5.2a due to the more general formulation for the viscous stress in the momentum equation for the dispersed phase. The behavior in the old formulation may be reproduced by dividing the dispersed phase viscosity, μ_D, by the variable `ee.phidPos`.

Backward Compatibility with Version 4.3b

BUBBLE-INDUCED TURBULENCE IN BUBBLY FLOW

Equation terms accounting for bubble-induced turbulence in the Turbulent Bubbly Flow interface are now added correctly. As a result, models with Turbulent Bubbly Flow now show higher levels of turbulence.

EULER-EULER MODEL, LAMINAR FLOW

The results from models using the Euler-Euler Model, Laminar Flow interface may differ in version 5.2a due to the more general formulation for the viscous stress in the momentum equation for the dispersed phase. The behavior in the old formulation may be reproduced by dividing the dispersed phase viscosity, μ_D, by the variable `ee.phidPos`.

Backward Compatibility with Version 4.3a

MIXTURE MODEL EQUATION FORMULATION

The Mixture Model equation formulation has been revised to increase the accuracy and facilitate the solution of a number of problem setups. As a consequence of the revision, old models can return slightly different results in the new version than they did in previous versions. As part of the revision, a penalty diffusion on the dispersed phase has been deactivated, because after the other revisions, it made the equation system too nonlinear and therefore more difficult to converge. However, you can still activate the penalty diffusion in the **Advanced** section of the Mixture Model physics interface settings.

EULER-EULER MODEL, LAMINAR FLOW

The results from models using the Euler-Euler Model, Laminar Flow interface may differ in version 5.2a due to the more general formulation for the viscous stress in the momentum equation for the dispersed phase. The behavior in the old formulation may be reproduced by dividing the dispersed phase viscosity, μ_D, by the variable `ee.phidPos`.

Backward Compatibility with Version 4.3

FLUID-STRUCTURE INTERACTION

The Fluid-Structure Interaction (FSI) multiphysics interface has been updated. The separate `vWall` field is no longer required and has been removed. FSI models from 4.3 and earlier versions that include Time Dependent study steps will be affected in the following ways:

- Model files for Java® will fail. Any reference to the `vWall` field must be removed.
- Models older than 4.3 (that is, 4.2a and older) must clear their time-dependent solutions when opened in 5.2a. An alternative procedure is to:
    - Open the model in version 4.3.
    - Right-click any study with a time-dependent study step and select *Update Solution*.
    - Save the model.
    - Open the model in version 5.2a.

FSI models with only stationary study steps will not be affected. Note that `vWall` will still be available as a variable. Hence, references to `fsi.vWall` made in, for example, another physics interface still work.

BRINKMAN EQUATIONS AND FREE AND POROUS MEDIA FLOW

The momentum equations in the Brinkman Equations interface and the Free and Porous Media Flow interface have been corrected. The term $-Q_{\mathrm{br}}\cdot\mathbf{u}/\varepsilon_\mathrm{p}^2$ in the right-hand side previously lacked the factor $1/\varepsilon_\mathrm{p}^2$, where ε_p is the porosity.

REACTING FLOW INTERFACE NAME CHANGE

The Reacting Flow interfaces have been renamed Reacting Flow in Porous Media. If you open a model using either the Reacting Flow, Diluted Species or the Reacting Flow, Concentrated Species interface, the interface is renamed Reacting Flow in Porous Media.

WEAK CONSTRAINTS UPDATE FOR FLUID FLOW INTERFACES

The weak constraint formulations for the following boundary conditions in the following interfaces have been updated:

Laminar Flow and Creeping Flow

- Symmetry
- The Slip boundary condition in the Wall feature

Turbulent Flow, k-ε and Turbulent Flow, k-ω

- Symmetry
- The Wall functions boundary condition in the **Wall** feature
- The Moving Wall (wall functions) boundary condition in the **Wall** feature
- The Slip boundary condition in the **Wall** feature

Turbulent Flow, low-Re k-ε and Turbulent Flow, Spalart-Allmaras

- Symmetry
- The Slip option in the **Wall** feature

Rotating Machinery

- Symmetry
- The Slip boundary condition in the **Wall** feature
- The Wall Functions boundary condition in the **Wall** feature (turbulent flow only)

- The Moving Wall (wall functions) boundary condition in the **Wall** feature (turbulent flow only)
- Rotating Wall (turbulent flow only)

Bubbly Flow

- Symmetry
- The Slip boundary condition for the liquid phase in the **Wall** feature
- The Wall Functions boundary condition for the liquid phase in the **Wall** feature (turbulent flow only)

Mixture Model

- Symmetry
- The Slip boundary condition for the mixture in the **Wall** feature
- The Wall Functions boundary condition for the mixture in the **Wall** feature (turbulent flow only)

Euler-Euler Model, Laminar Flow

- Symmetry
- The Slip boundary condition for either phase in the **Wall** feature
- The Slip boundary condition for either phase in the **Inlet** feature

Brinkman Equations and Free and Porous Media Flow

- Symmetry
- The Slip boundary condition in the **Wall** feature

Two-Phase Flow, Level Set

- Symmetry
- The Slip boundary condition in the **Wall** feature
- The Wetted Wall boundary condition in the **Wall** feature
- The Moving Wetted Wall boundary condition in the **Wall** feature
- The Wall Functions boundary condition in the **Wall** feature (turbulent flow only)
- The Moving Wall (wall functions) boundary condition in the **Wall** feature (turbulent flow only)

Two-Phase Flow, Phase Field

- Symmetry
- The Slip boundary condition in the **Wall** feature

- The Wall function boundary condition in the **Wall** feature (turbulent flow only)
- The Moving Wall (wall functions) boundary condition in the **Wall** feature (turbulent flow only)

Nonisothermal Flow and Conjugate Heat Transfer

- Symmetry
- The Slip boundary condition in the **Wall** feature
- The Wall Functions boundary condition in the **Wall** feature (turbulent flow k-ε and turbulent flow k-ω only)
- The Moving Wall (wall functions) boundary condition in the **Wall** feature (turbulent flow k-ε and turbulent flow k-ω only)

High Mach Number Flow

- Symmetry
- The Slip boundary condition in the **Wall** feature
- The Wall Functions boundary condition in the **Wall** feature (turbulent flow k-ε only)
- The Moving Wall (wall functions) boundary condition in the **Wall** feature (turbulent flow k-ε only)

These boundary conditions are now formulated using the same set of Lagrange multipliers as all of the other boundary conditions for the dependent velocity variables. The previously used Lagrange multiplier `un_lm` has been removed.

When you open models saved in version 4.3, they include `un_lm` until the model is re-solved. In some cases, occurrences of `un_lm` in the solver sequence must be replaced manually. This is the case if `un_lm` was the only Lagrange multiplier component in a segregated group or the only Lagrange multiplier component of a Vanka smoother. Alternatively, you can generate a new automatic solver sequence. Models saved in versions prior to version 4.3 must either be re-solved in version 5.2a for postprocessing, or opened and re-saved in version 4.3 before being opened in version 5.2a.

Weak constraints for the **Interior Wall** feature are no longer available.

REVISION OF THE TURBULENCE MODELS

The formulations of some variables in the turbulence models have been revised in order to improve accuracy. Models using a turbulence model can display a different convergence behavior in version 5.2a than in version 4.3 and the results can differ slightly between the versions.

EULER-EULER MODEL, LAMINAR FLOW

The results from models using the Euler-Euler Model, Laminar Flow interface may differ in between versions 4.3 and 5.2a due to the more general formulation for the viscous stress in the momentum equation for the dispersed phase. The behavior in the old formulation may be reproduced by dividing the dispersed phase viscosity, μ_D, by the variable `ee.phidPos`.

Chemical Reaction Engineering Module

New Functionality in Version 5.3a

THERMODYNAMICS

New built-in functionality in the Thermodynamics node is available under Global Definitions, or from the Reaction Engineering toolbar when using the same interface. Two types of nodes for chemical species properties are now available in Thermodynamics: Property Package and External Property Package. The Property Package node includes a built-in database containing transport and thermodynamic properties of 251 chemical species. Built-in machinery is used for computing property functions, for both pure species and mixtures, based on the underlying database and a wide range of implemented thermodynamic models. The External Property Package node is used to define property functions in the same manner, but requires CAPE-OPEN compliant packages from an external software provider. The Property package node contains all new functionality, while the External Property Package node corresponds to what was previously available in the Thermodynamics node.

The Reaction Engineering interface and the Chemistry interface support coupling to a Thermodynamics property package. When coupled, the species and mixture properties required by the interface are automatically defined and added to the corresponding property package. Coupling is supported both for Property Package nodes and External Property Package nodes.

IMPROVED DIFFUSION MODELING

Improved diffusion model implementations are available in the Transport of Concentrated Species interface. Using the Mixture-averaged and Fick’s Law diffusion models, the molecular flux can now be formulated in terms of the mole fraction gradient or mass fraction gradient. A diffusion model correction has also been implemented. This correction ensures that the net diffusion, with respect to the mass-averaged velocity, is zero.

New Models in Version 5.3a

Models showcasing the new functionality:

A Thermodynamics folder has been added to the Application Library of the Chemical Reaction Engineering Module. The folder contains two new tutorial models using the new built-in functionality: *Engine Coolant Properties* and *Distillation Column*. It also contains the tutorial model *Hydrodealkylation Membrane Reactor*, which has been updated to use the new Property Package feature.

Backward Compatibility with Version 5.0 and Earlier

REACTION ENGINEERING AND CHEMISTRY INTERFACES

The API syntax for creating surface species has changed. Running Java® files from previous versions may fail due to the new default name indexes, including **ads** for surface species and **s** for solids in the Reaction Engineering and Chemistry interfaces. The surface species names and solid species names need to be changed accordingly in any Java® code:

- Change any species index `(s)` to `(ads)`
- Change any species index `(S)` to `(s)`

The following shows an example of the needed code change:

Version 5.0 and earlier:

```
model.physics("re").feature("rch1").set("formula", "A+B(S)=>C+D(s)");
model.physics("re").feature("spec1").set("specName", "As(s)");
model.physics("chem").feature("rch1").set("formula", "A+B(S)=>C+D(s)");
model.physics("chem").feature("spec1").set("specName", "As(s)");
```

Change this to

Version 5.2 and later:

```
model.physics("re").feature("rch1").set("formula", "A+B(s)=>C+D(ads)");
model.physics("re").feature("spec1").set("specName", "As(ads)");
model.physics("chem").feature("rch1").set("formula", "A+B(s)=>C+D(ads)");
model.physics("chem").feature("spec1").set("specName", "As(ads)");
```

TRANSPORT OF CONCENTRATED SPECIES INTERFACE

Updated Internal Variable Names

The internal variable names of vector and tensor components have changed in the *Transport of Concentrated Species* interface. In most cases, users will not be affected by this change. When opening a model file created in a previous version, the previous variable definitions remain. However, when updating or recomputing the solution, new variable names will be generated. This means that if any of these variable names have been used in a user-defined expression or plot, they need to be updated manually as described below.

In version 5.1, the spatial direction indicator string (x, xy, and so on) has been moved to the end of the variable name. This applies to all vector and tensor variables generated by the interface.

For example, the diffusive flux vector components have changed (in 5.0 and earlier) from

```
tcs.dfluxx_w1, tcs.dfluxy_w1, tcs.dfluxz_w1
```

to (in 5.2 and later)

```
tcs.dflux_w1x, tcs.dflux_w1y, tcs.dflux_w1z
```

and the components of the diffusion tensor, defined by the Fick's Law diffusion model, have been changed (in 5.0 and earlier) from

```
tcs.Dfxx_w1, tcs.Dfxy_w1, tcs.Dfxz_w1, tcs.Dfyx_w1, ...
```

to (in 5.2 and later)

```
tcs.Df_w1xx, tcs.Df_w1xy, tcs.Df_w1xz, tcs.Df_w1yx, ...
```

Java® API Syntax Changes

The API syntax for setting physics properties has been changed in the Transport of Concentrated Species interface. Now the property name corresponds to the name of the GUI section containing the property. In order to run Java® files from previous versions, apply the syntax changes as given in the examples below:

Version 5.0 and earlier:

```
model.physics("tcs").prop("DiffusionModel").setIndex("DiffusionModel",
"MaxwellStefan", 0);
model.physics("tcs").prop("DiffusionModel").setIndex("DiffusionModel",
"MixtureAveraged", 0);
model.physics("tcs").prop("DiffusionModel").setIndex("DiffusionModel",
"FicksLaw", 0);
model.physics("tcs").prop("Convection").setIndex("Convection", "0", 0);
model.physics("tcs").prop("Migration").setIndex("Migration", "0", 0);
model.physics("tcs").prop("Regularization").setIndex("Regularization", "1",
0);
```

For version 5.2 and later, change this to:

```
model.physics("tcs").prop("TransportMechanism").set("DiffusionModel",
"MaxwellStefan");
model.physics("tcs").prop("TransportMechanism").set("DiffusionModel",
"MixtureAveraged");
```

```
model.physics("tcs").prop("TransportMechanism").set("DiffusionModel",
"FicksLaw");
model.physics("tcs").prop("TransportMechanism").set("Convection", false);
model.physics("tcs").prop("TransportMechanism").set("Migration", false);
model.physics("tcs").prop("AdvancedSettings").set("Regularization", "0");
```

SURFACE REACTIONS INTERFACE

Updated Internal Variable Names

The same pattern for vector and tensor variable name changes, as described in the Transport of Concentrated Species Interface section above, holds true for the Surface Reactions interface in version 5.2a.

Java® API Syntax Changes

The API syntax for setting the diffusion tensor of surface species, the molar mass and density of the bulk species, and initial values have been changed in the Surface Reactions interface.

Setting Diffusion Tensor

The surface species name needs to be included in the first argument of the set method. In order to run Model Java® files from previous versions, apply the syntax changes as shown in the example below:

Version 5.0 and earlier:

```
model.physics("sr").feature("sp1").set("D_0", new String[]{"Dcs1", "0", "0",
"0", "Dcs1", "0", "0", "0", "Dcs1"});
model.physics("sr").feature("sp1").set("D_1", new String[]{"Dcs2", "0", "0",
"0", "Dcs2", "0", "0", "0", "Dcs2"});
```

For version 5.2 and later, change this to:

```
model.physics("sr").feature("sp1").set("D_cs1", new String[]{"Dcs1", "0",
"0", "0", "Dcs1", "0", "0", "0", "Dcs1"});
model.physics("sr").feature("sp1").set("D_cs2", new String[]{"Dcs2", "0",
"0", "0", "Dcs2", "0", "0", "0", "Dcs2"});
```

Setting Molar Mass and Density

The bulk species name needs to be included in the first argument of the `setIndex` method. Apply the Java® API syntax changes as shown in the example below:

Version 5.0 and earlier:

```
model.physics("sr").feature("sp1").set("M_0", "Mcb1");
model.physics("sr").feature("sp1").set("M_1", "Mcb2");

model.physics("sr").feature("sp1").set("rho_0", "rhocb1");
model.physics("sr").feature("sp1").set("rho_1", "rhocb2");
```

For version 5.2 and later, change this to:

```
model.physics("sr").feature("sp1").setIndex("M_cb1", "Mcb1", 0);
model.physics("sr").feature("sp1").setIndex("M_cb2", "Mcb2", 0);

model.physics("sr").feature("sp1").setIndex("rho_cb1", "rhocb1", 0);
model.physics("sr").feature("sp1").setIndex("rho_cb2", "rhocb2", 0);
```

Setting Initial Values

Apply the Java® API syntax changes as shown in the example below:

Version 5.0 and earlier:

```
model.physics("sr").feature("init1").set("cs1", "cs10");
model.physics("sr").feature("init1").set("cs2", "cs20");
model.physics("sr").feature("init1").set("cb1", "cb10");
model.physics("sr").feature("init1").set("cb2", "cb20");
```

For version 5.2 and later, change this to:

```
model.physics("sr").feature("init1").setIndex("initcs", "cs10", 0);
model.physics("sr").feature("init1").setIndex("initcs", "cs20", 1);
model.physics("sr").feature("init1").setIndex("initcb", "cb10", 0);
model.physics("sr").feature("init1").setIndex("initcb", "cb20", 1);
```

Corrosion Module

New Functionality in Version 5.3a

The following new functionality is available:

- Improved Electrolyte Current boundary node in the Current Distribution, Shell interface (cdshell interface).
- New Thin Electrode Surface boundary node (in the Primary Current Distribution; Secondary Current Distribution; and Tertiary Current Distribution, Nernst-Planck interfaces).
- Ohm's law has been added as an Electric Potential model option in the Sacrificial Edge Anode node. A new Wiring Edges subnode has been added to the Sacrificial Edge Anode when using Ohm's law (in the Primary Current Distribution and Secondary Current Distribution interfaces).
- Sacrificial Edge Anode now considers symmetry boundaries (in the Primary Current Distribution and Secondary Current Distribution interfaces).
- A new Electrochemical Heating Multiphysics node automatically couples both boundary and domain heat sources and temperature between an Electrochemistry branch interface and a Heat Transfer interface
- A new Electrode Current boundary node has been added to the Electrode Shell interface (els).
- The Reactions node used on Porous Electrode domains in the Tertiary Current Distribution, Nernst-Planck interfaces now adds source terms for the electrode potential.
- New units: Wh and Ah.
- Dissolving-depositing species concentrations are no longer solved for by default for stationary studies.

New and Updated Models in Version 5.3a

New *Stress Corrosion* and *Stray Current Pipeline Corrosion* models have been added to the Corrosion Module Application Library.

The *Corrosion Protection of a Ship Hull* model in the Corrosion Module Application Library has been updated to use the Thin Electrode Surface boundary node.

Backward Compatibility with Version 5.3

The default tags of the Secondary and Tertiary Current Distribution interfaces have been changed to `cd` (old tag was `siec`) and `tcd` (old tag was `tcdee`), respectively. When running the Java API, an explicit command to set the tag of the interface to the old default tag may have to be added to get Java API files created prior to 5.3a to run. For example: `model.component("comp1").physics("siec").tag("siec");`

Backward Compatibility with Earlier Versions

SURFACE REACTIONS INTERFACE

See Chemical Reaction Engineering Module.

ECAD Import Module

New Functionality in Version 5.3a

- Support has been added to import printed circuit board (PCB) files of the IPC-2581 format.
- When importing GDSII files, it is now possible to split layers into separate layers based on the datatype setting of the objects on the layer.

Backward Compatibility with Version 5.2a

- The default value for the `sellayer` property of the `Import` function is now `on`. Previously, the default was `off`.
- The default value for the `sellayershow` property of the `Import` function is now `all`. Previously, the default was `dom`.

Backward Compatibility with Version 5.0

- The default value for the `grouping` property of the `Import` function is now `layer`. Previously, the default was `all`.
- For ODB++® and ODB++(X) files (the `type` property is `odb`), new rules apply for the initialization of the `importlayer` property of the `Import` function. Now, only layers of the types *Metal* and *Dielectric* are initialized with the string `on` in the `importlayer` string array. Previously, this also included layers of the type *Drill*.

Electrochemistry Module

New Functionality in Version 5.3a

The following new functionality is available:

- A new Electrochemical Heating multiphysics node automatically couples both boundary and domain heat sources and temperature between an Electrochemistry branch interface and a Heat Transfer interface.
- A new Electrode Current boundary node has been added to the Electrode Shell interface (els).
- The Reactions node used on Porous Electrode domains in the Tertiary Current Distribution, Nernst-Planck interfaces now adds source terms for the electrode potential.
- New units: Wh and Ah.

Backward Compatibility with Version 5.3

The default tags of the Secondary and Tertiary Current Distribution interfaces have been changed to `cd` (old tag was `siec`) and `tcd` (old tag was `tcdee`), respectively. When running the Java API, an explicit command to set the tag of the interface to the old default tag may have to be added to get Java API files created prior to 5.3a to run. For example: `model.component("comp1").physics("siec").tag("siec");`

Backward Compatibility with Previous Versions

SURFACE REACTIONS INTERFACE

See Chemical Reaction Engineering Module.

Electrodeposition Module

New Functionality in Version 5.3a

The following new functionality is available:

- New Thin Electrode Surface boundary node (in the Primary Current Distribution; Secondary Current Distribution; and Tertiary Current Distribution, Nernst-Planck interfaces).
- The Level-Set interface is now included in the Electrodeposition Module.
- A new Electrochemical Heating multiphysics node automatically couples both boundary and domain heat sources and temperature between an Electrochemistry branch interface and a Heat Transfer interface.
- A new Electrode Current boundary node has been added to the Electrode Shell interface (els).
- The Reactions node used on Porous Electrode domains in the Tertiary Current Distribution, Nernst-Planck interfaces now adds source terms for the electrode potential.
- New units: Wh and Ah.
- Dissolving-depositing species concentrations are no longer solved for by default for stationary studies.

New and Updated Models in Version 5.3a

New *Aluminum Anodization* and *Copper Deposition in a Trench using the Level Set Method* models have been added to the Electrodeposition Module Application Library.

The *Electrocoating of a Car Door* model in the Electrodeposition Module Application Library has been updated to use the Thin Electrode Surface node.

Backward Compatibility with Version 5.3

The default tags of the Secondary and Tertiary Current Distribution interfaces have been changed to `cd` (old tag was `siec`) and `tcd` (old tag was `tcdee`), respectively. When running the Java API, an explicit command to set the tag of the interface to the old default tag may have to be added to get Java API files created prior to 5.3a to run. For example: `model.component("comp1").physics("siec").tag("siec");`

Backward Compatibility with Previous Versions

SURFACE REACTIONS INTERFACE

See Chemical Reaction Engineering Module.

Fatigue Module

Backward Compatibility with Version 5.3

Several of the fatigue laws for low cycle fatigue are customarily expressed in terms of number of load reversals.

In previous versions, the distinction between *number of load reversals* to failure and *number of load cycles* to failure has not been clear, neither in the theory sections, nor in result output. In version 5.3a, the result is always in terms of cycles, and the documentation has been clarified at a number of places.

The affected fatigue models are **Basquin**, **Coffin-Manson**, **Combined Basquin and Coffin-Manson**, **Morrow**, **Smith-Watson-Topper (SWT)**, **Wang-Brown**, and **Fatemi-Socie**.

Geomechanics Module

New Functionality in Version 5.3a

ELASTOPLASTIC SOIL MATERIAL MODELS

This new family of soil material models is characterized by the fact that the elastic and plastic behaviors are not independent.

One such material model existed in previous versions: **Cam-Clay**. It has now been improved and renamed to **Modified Cam-Clay**. It has also been moved into the new group **Elastoplastic Soil Material Models** together with three new material models commonly used in the geomechanics community: **Hardening Soil**, **Extended Barcelona Basic**, and **Modified Structured Cam-Clay**.

When you open a model created in an earlier version that contains a **Cam-Clay** material, this node is automatically transformed into a corresponding **Modified Cam-Clay** material.

The pore pressure is no longer available as an input field. Instead, an **External Stress** subnode is used for this purpose, and this node is added automatically.

The check box **Nearly incompressible material** present in the old **Cam-Clay** material has been removed in **Modified Cam-Clay**, since the soils to which this material model is applicable do not require a mixed formulation due to their compressibility.

IMPROVED DEFAULT PLOTS

Contours showing plastic strains and creep strains are now added to the default stress plot when such materials are used.

Heat Transfer Module

New and Improved Functionality in Version 5.3a

THERMAL PERTURBATION, EIGENFREQUENCY STUDY

The Heat Transfer interfaces now support the **Eigenfrequency** solver, which computes eigenmodes and eigenfrequencies of linearized models around an equilibrium state. The predefined **Thermal perturbation, Eigenfrequency domain** solver sequence first solves a **Stationary** study step to find the equilibrium state and then reuses that solution as a starting point for the second **Eigenfrequency** step. The usage of this study is illustrated in the Thermoelastic damping in a MEMS Resonator tutorial of the MEMS Module.

INFLOW BOUNDARY CONDITION

The new **Inflow** boundary condition is available at inlet boundaries where it is suitable to define the temperature and inlet accounting for upstream temperature and pressure conditions. It is applicable on the exterior boundaries of a fluid domain. This condition accounts for the backward effect of the model on the upstream temperature based on the flow velocity. Hence the temperature at walls where the flow velocity is zero is not constrained to the upstream temperature. The **Inflow Heat flux** boundary condition becomes obsolete and its behavior can be reproduced by applying both **Inflow** and **Boundary Heat Source** conditions.

MOISTURE FLOW

The new **Moisture Flow** multiphysics interfaces are available under the **Chemical Species Transport** branch to model moisture transport in air by laminar and turbulent flows. The different interfaces couple the laminar and turbulent versions of the Single-Phase Flow interfaces with the Moisture Transport in Air interface. The **Moisture Flow** multiphysics coupling node handles turbulent mixing and moisture wall functions for turbulent flows, and accounts for moisture content in the fluid flow equations in air. Note that it can be used in combination with the **Nonisothermal Flow** and **Heat and Moisture** multiphysics couplings to obtain a coupled heat and moisture transport modeling in moist air.

THERMAL DAMAGE EXTENDED CAPABILITIES

The **Irreversible Transformation** attribute, available under the **Solid** node and the **Biological Tissue** node (with the **Include damage integral analysis** check box selected), has a new **User defined** option for the transformation model, and the **Energy absorption** option has been renamed **Arrhenius kinetics**. In addition, it is now possible to set the order of the polynomial Arrhenius equation with this latter option. All the **Transformation model** options allow you to account for the generation or loss of heat in the energy balance during transformation.

MODELING OF THERMAL CONTACT BY AN EQUIVALENT THIN RESISTIVE LAYER

A new **Equivalent thin resistive layer** option is available for the contact model in the **Thermal Contact** boundary condition between two solid domains. This option offers three possibilities to define the layer conductance, by specifying either the **Layer conductance**, **Layer resistance**, or **Layer thermal conductivity** and **Layer thickness**.

EXTENDED FLUID OPTIONS FOR CONVECTIVE HEAT FLUX

In the **Heat Flux** boundary condition, two new options are available for the external **Fluid** when using heat transfer coefficient correlations for convective heat transfer: **Moist air** and **From material**. With the **Moist air** option, you can account for moisture content in the Nusselt correlations by setting the **External relative humidity** and the **Surface relative humidity**. When the **From material** option is selected, you can use any material available on the corresponding boundary in the **Materials** node.

In addition, the **Absolute pressure** input is no longer displayed for the **Water** and **Transformer oil** options, as it is not used in the correlations.

IMPROVED CAPABILITIES FOR HEAT AND MOISTURE TRANSPORT

The **Moisture Flux** node of the Moisture Transport interface has extended capabilities for the definition of moisture flux on boundaries.

- Convective moisture flux: the **Convective moisture flux** option has been renamed **Convective moisture flux, pressures difference**, and a new **Convective moisture flux, concentrations difference** has been added. With this latter option, it is possible to use heat and mass boundary layer analogy to define the moisture transfer coefficient. Therefore, in addition to the **User defined** option, all the Nusselt correlations defined for convective heat transfer in various configurations (**External natural convection**, **Internal natural convection**, **External forced convection**, and **Internal forced convection**) are also available. With the **From heat and moisture transport analogy** option, it is possible to select any heat transfer coefficient defined by a **Heat Flux** boundary condition in a Heat Transfer interface, or alternatively to define a **User defined** value, and to apply the analogy.

- Contribution to evaporation flux: the new **Contributes to evaporation flux** check box in the **Evaporation** section allows you to account for the moisture flux in latent heat source on boundaries.

The **Heat and Moisture** multiphysics coupling node now handles the latent heat sources due to evaporation and condensation on surfaces. This behavior can be controlled through the **Include latent heat source on surfaces** check box in the **Latent Heat** section. When selected, the latent heat sources computed by the **Moist Surface**, **Wet Surface**, and **Moisture Flux** nodes are included as heat sources in the Heat Transfer interface.

RADIATIVE BEAM IN ABSORBING MEDIA INTERFACE

The new Radiative Beam in Absorbing Media physics interface can be found under the **Radiation** group of the **Heat Transfer** branch. It provides features to model the propagation and absorption of high-intensity radiative beams, like lasers, when they penetrate a partially transparent material, depositing power into the material itself. The **Medium Properties** domain node defines the absorbing medium properties and the radiative heat source due to absorption. Multiple incident beams can be defined with the **Incident Intensity** boundary node. Additionally, the **Opaque Surface** boundary node specifies a wall that absorbs all the radiative intensity, producing heat, and the **Transparent Surface** node can be applied on boundaries that let the radiative intensity exit without depositing energy.

TIME-DEPENDENT CLIMATE DATA IMPROVEMENTS

The latest version of ASHRAE climate database, Weather Data Viewer version 6.0(c), is now available for defining ambient variables in the **Ambient Settings** section of the Heat Transfer interfaces. Monthly and hourly averaged measurements, listed in the ASHRAE 2017 handbook by the American Society of Heating, Refrigerating, and Air-Conditioning Engineers (ASHRAE), are available for about 8000 weather stations worldwide. In addition, when the **Weather station** is searched **From list**, the new **Set Weather Station** button allows you to easily browse and search the list of stations in the **Weather Station** dedicated browser.

HEAT TRANSFER IN SHAPE MEMORY ALLOYS

The new **Shape Memory Alloy** domain node is available in the Heat Transfer interfaces when the **Heat transfer in alloys** check box is selected in the **Physical Model** section. It accounts for the Martensite and Austenite volume fraction in alloys to define effective thermal properties from the thermal properties of each phase.

It is designed to be combined with the **Shape Memory Alloy** feature from the Structural Mechanics interface, which defines the Martensite volume fraction and the energy absorption or release due to the Austenite/Martensite structural changes.

SHELLS VERSION OF THE GEOMETRY PARTS FOR HEAT SINKS

The Part Library in the Heat Transfer Module has been updated to include shell versions of the different parameterized geometry parts dedicated to heat sinks with pin fins, straight fins, or pin fins with dissimilar dimensions on the borders. The new `shell` parameter allows you to define the fins as solids or boundaries for computational efficiency. When `shell=1`, the `X_fins_bottom`, `Y_fins_bottom`, `X_fins_top`, and `Y_fins_top` parameters are not used, and the step and bottom fillet transformations are disabled.

ELECTROMAGNETIC HEATING MULTIPHYSICS COUPLING

The new **Electromagnetic Heating** multiphysics coupling replaces the **Electromagnetic Heat Source**, **Boundary Electromagnetic Heat Source**, and **Temperature Coupling** multiphysics couplings, by automatically coupling both boundary and domain heat sources and temperature between an AC/DC branch interface and a Heat Transfer interface. It contains **Domain Selection** and **Boundary Selection** sections to specify where to include the electromagnetic volumetric and surface losses in the heat equation. The **Electromagnetic Heat Source** and **Boundary Electromagnetic Heat Source** coupling nodes become obsolete.

THERMOELECTRIC EFFECT MULTIPHYSICS COUPLING

The **Thermoelectric Effect** multiphysics coupling now accounts for thermoelectric effects both in domains and on boundaries where electrical and thermal models are defined. It contains a **Domain Selection** and a **Boundary Selection** section to specify where to include the heat sources or sinks in the heat equation. It is the default feature of the Thermoelectric Effect multiphysics interface, along with the new **Electromagnetic Heating** multiphysics coupling node. The **Boundary Thermoelectric Effect** coupling node becomes obsolete.

MISCELLANEOUS

- The stabilization has been updated for flow and temperature equations in time-dependent analyses. The new version of the stabilization is designed to improve robustness for transient simulation with small time steps. The new formulation introduces new shape functions, `ht.dt2Inv_T` for heat transfer and `spf.dt2Inv` for fluid flow, that are placed in a **Previous Solution** node of the study with the default solver settings. Models created before version 5.3a still use the former stabilization.

- In the **Thermal Contact** boundary node, it is now possible to set specific upside and downside Young's moduli and Poisson's ratios for the **Mikic elastic correlation** option.
- In the **Fluid** feature with **Fluid type** set to **Moist air**, as well as the **Moist air** feature, the dry air properties used to compute the moist air properties are now taken from the **Built-in** material database. In addition, the ideal law for gas now uses the `R_const` variable instead of a local definition.
- Sketches have been added to illustrate the different configurations available in the **Equivalent conductivity for convection** section of the **Fluid** domain node. A new **Sketch** section is displayed just above the **Equivalent conductivity for convection** section when the **Equivalent conductivity for convection** check box is selected.
- The **Porous Medium** and **Building Material** domain features are now compatible with **Infinite Elements**.
- The account for fluid mass sources (defined in **Darcy's Law** interface, for example) in the postprocessing variables for heat balance has been improved in multiphysics couplings involving heat transfer and porous flow.
- The definition of the postprocessing variables `ndflux_u` and `ndflux_d` for heat balance has been improved for the **Diffuse Surface** feature. It now correctly accounts for the radiative flux at interior boundaries.

New Applications in Version 5.3a

BUOYANCY FLOW IN AIR

This new tutorial model studies the stationary state of free convection in a cavity filled with air and bounded by two vertical plates. The two plates are maintained at different temperatures, inducing a buoyancy flow in the air domain. Operating conditions are defined so that the flow regime is laminar. Note that the model is built in a similar manner as the already existing Buoyancy Flow in Water model. The main difference between the two models is that air density, modeled using the ideal gas law, is temperature and pressure dependent.

LAMINAR NONISOTHERMAL FLOW IN A CIRCULAR TUBE

This new tutorial model computes the velocity, pressure, and temperature distribution in a circular tube using a 2D axisymmetry geometry. The operating conditions correspond to a nonisothermal laminar flow. The heat transfer coefficient deduced from the simulation is in good agreement with the one based on the published Nusselt number from experimental measurements.

TURBULENT NONISOTHERMAL FLOW OVER A FLAT PLATE

This new tutorial model computes the velocity, pressure, and temperature distribution over a plate. Once the flow is turbulent and fully developed, it reaches a hot region of the plate. The heat transfer coefficient between the air flow and the plate has been measured experimentally and different Nusselt-number-based correlation are available. These simulation results are in good agreement with the published data.

DYNAMIC WALL HEAT EXCHANGER

This new tutorial model shows a compact heat exchanger that has enhanced performance thanks to the use of a deforming wall with an oscillating wave shape. The wall oscillations induce mixing in the fluid and reduce the formation of thermal boundary layers. Additionally, the wave-shaped deformation induces a pumping effect similar to the peristaltic pumping, which mitigates the pressure losses. This model includes the Conjugate Heat Transfer multiphysics coupling and Moving Mesh features to handle the wall and channel deformation. The pressure drop across the heat exchanger and the overall heat transfer coefficient are computed for static and dynamic heat exchangers.

Updated Applications in Version 5.3a

The following updates have been made in the tutorials and applications to take advantage of the new functionalities:

- Several applications have been updated to use the new **Inflow** boundary condition instead of the **Temperature** boundary condition at inlet boundaries, to get a more accurate and realistic physical model.
- The Evaporative Cooling of Water tutorial has been updated to use the new **Moisture Flow** multiphysics coupling node and the handling of the latent heat source due to evaporation at the top of the glass by the **Heat and Moisture** coupling node.
- The Electronic Chip Cooling tutorial has been updated to use the new **Equivalent thin resistive layer** option of the **Thermal Contact** boundary node.
- Several applications have been updated to use the new **Electromagnetic Heating** and **Thermoelectric Effect** multiphysics coupling nodes.

Backward Compatibility with Version 5.3

- The **Electromagnetic Heat Source** and **Boundary Electromagnetic Heat Source** coupling features are now obsolete. They have been merged into a single **Electromagnetic Heating** coupling feature. When opening a model created in version 5.3 or earlier in the GUI, the **Electromagnetic Heat Source** and **Boundary Electromagnetic Heat Source** features are automatically migrated to **Electromagnetic Heating** features with selections (respectively a domain and a boundary selection) that match the selections of the migrated features. The obsolete **Electromagnetic Heat Source** and **Boundary Electromagnetic Heat Source** features remain available through the COMSOL API so that existing scripts remain compatible. However, it is recommended to update the scripts to use the new **Electromagnetic Heating** feature as the obsolete feature will be discontinued in future versions. Note that temperature coupling is enforced in the 5.3a version of the **Electromagnetic Heating** coupling feature.
- The **Thermoelectric Effect** and **Boundary Thermoelectric Effect** coupling features have been merged into a new **Thermoelectric Effect** coupling feature. When opening a model created in version 5.3 or earlier in the GUI, the **Thermoelectric Effect** and **Boundary Thermoelectric Effect** features are automatically migrated to the new **Thermoelectric Effect** feature with selections (respectively a domain and a boundary selection) that match the selections of the migrated features. The obsolete **Thermoelectric Effect** and **Boundary Thermoelectric Effect** features remain available through the COMSOL API so that existing scripts remain compatible. However, it is recommended to update the scripts to use the new **Thermoelectric Effect** feature as the obsolete feature will be discontinued in the future. Note that temperature coupling is enforced in the 5.3a version of the **Thermoelectric Effect** coupling feature.
- The **Inflow Heat Flux** boundary condition is now obsolete. It can be replaced by the new **Inflow** boundary condition and the **Boundary Heat Source** condition.
- The **Enable conversions between material and spatial frame** check box, displayed in the **Advanced Settings** section of the Heat Transfer interfaces when the model component supports moving mesh, has been removed. This means that conversion between material and spatial frames is always effective when these frames differ.
- The **Energy absorption** option has been renamed **Arrhenius kinetics** in the **Transformation model** list of the **Irreversible Transformation** and **Biological Tissue** nodes.
- The **Convective moisture flux** option has been renamed **Convective moisture flux, pressures difference** in the **Moisture Flux** node.

- The `rhoInt`, `CpInt`, and `gammaInt` variables, defined by the **Fluid** domain feature, have been removed.
- The `q_evap` variable, defined by the **Moist Surface** and **Wet Surface** boundary nodes, has been replaced by the `q_evaptot` variable.

The **General** option for **Thin film model** has been removed from the **Thin Film** fallback node.

Backward Compatibility with Version 5.2a

The obsolete features **Convective Heat Flux** and **Out-of-plane Convective Heat Flux** are removed in Version 5.3. These were marked as obsolete with a warning message since Version 5.0. These features will be removed from your model when opening in Version 5.3. Use instead the corresponding **Heat flux** and **Out-of-Plane Heat Flux** features, with the option **Convective heat flux**. The variables `ht.chflux`, `ht.chflux_u`, `ht.chflux_d`, and `ht.chflux_z` defined by these features have been also removed from equation view and post-processing variables menu.

The new default option in **Thin Moisture Barrier** feature is **Vapor barrier material properties**, and the **Moisture barrier material properties** option has been removed. When loading a MPH file that was created before version 5.3 this latter option will be cleared, and the corresponding user inputs used to define the **Moisture transfer coefficient** directly. However, for the API, it will be necessary to update the code with an existing option name.

In the **Equivalent Conductivity for Convection** section of the **Fluid** node, the new **Automatic** option for **Temperature difference** has become the default option. When loading a MPH file that was created before version 5.3, the **User defined** option will be set automatically with the previous value. However, for the API, it will be necessary to update the code to set the **User Defined** option.

Backward Compatibility with Version 5.2

Several enhancements or bug fixes may require an update of the existing applications or induce a change in the results. Here is the list of the changes:

- When **Solar position** is selected in **External Radiation Source**, the new option **Update time from solver** is selected by default. When loading an MPH file that was created before version 5.2a this option will be cleared. However, for the API, for time-dependent studies, it will be necessary to update the code to get the same behavior as in previous versions. In most cases, removing the t term in the Second field is the best option. Alternatively, clearing the **Update time from solver** option will ensure that the same behavior as before is obtained.

- The crosswind stabilization of the heat transfer and fluid flow interfaces has been updated. In general, the update reduces the smoothing induced by the stabilization compared to previous version. In addition, when you use the **Nonisothermal Flow** multiphysics coupling and the fluid is defined as incompressible, the uncoupled versions of the streamlines and crosswind stabilization are used.
- Since **Pair Thin Film** and **Pair Fracture** are now pair features, their selection needs to be redefined.
- The definition of the `ht.`*`feature`*`.fc(RH.T,pA)` function when **Moist air** fluid type is used in **Fluid** feature has been corrected. This change can affect the results in simulations where it is used.
- The frame support has been enhanced to support moving mesh combined with deformed geometry in the same component.
- The **Prescribed Radiosity** feature has been updated so that it no longer ignores the **Temperature** model input.
- In surface-to-surface boundary features, the ambient temperature definition has been fixed when the **Opacity Controlled** and **Defined ambient temperature on each side** options are used simultaneously.

Backward Compatibility with Version 5.1

CHANGES IN VARIABLE NAMES

The names of the heat capacity variables have changed in several features for consistency:

- In the **Bioheat** feature, `ht.cb` has become `ht.cp_b`.
- In the **Thin Layer** feature, `ht.cs` has become `ht.cp_s`.
- In the **Thin Rod** feature, `ht.cl` has become `ht.cp_l`.
- In the **Thin Layered Shell** feature, `htsh.cs1` has become `htsh.cp_s1`, and so on.
- In the **Heat Transfer in Porous Media** feature, `ht.c_pp` has become `ht.cp_p`, `ht.c_pp1` has become `ht.cp_p1`, and so on.
- In the **Immobile Fluids** feature, `ht.c_pg1` has become `ht.cp_g1`, and so on.
- In the **Fracture** feature, `ht.c_pfr` has become `ht.cp_fr`.

The names of the thermal resistance and absolute thermal resistance variables have changed in the **Isothermal Domain Interface** feature:

- `ht.idi1.Rs` has become `ht.idi1.R_t`
- `ht.idi1.Rsth` has become `ht.idi1.R_tth`

The name of the fluid viscosity variable has changed in the **Local Thermal Non-Equilibrium** multiphysics coupling: `ltne1.mu_f` has become `ltne1.mu`.

Backward Compatibility with Version 5.0

OBSOLETE FEATURES

A number of features are now obsolete because they have been merged or replaced by a more general features. However, when models from COMSOL 5.0 or before contain these features, some of the obsolete features are visible in version 5.2a with a warning sign notifying that the feature is now obsolete (see Table 1-1). Some features have been completely removed (see Table 1-2). A model containing any of them can be opened in 5.2a, but the feature will be missing.

Updating the models is recommended. Use the new feature and remove the obsolete feature to make sure that the model will remain compatible with future versions of COMSOL Multiphysics.

The table below summarizes the changes:

TABLE 1-1: OBSOLETE FEATURES THAT ARE OBSOLETE BUT THAN CAN BE LOADED IN 5.1

OBSOLETE FEATURE	REPLACE IN 5.2A BY
Convective Heat Flux	**Heat Flux** with the **Convective heat flux** option selected
Surface-to-Ambient Radiation	**Diffuse Surface** with **Include surface-to-surface radiation** deselected
Surface-to-Surface Radiation	**Diffuse Surface** with **Include surface-to-surface radiation** selected
Highly Conductive Layer	**Thin Layer** with **Layer type** set to **Conductive**
Thin Thermally Resistive Layer	**Thin Layer** with **Layer type** set to **Resistive**

TABLE 1-2: OBSOLETE FEATURES THAT ARE NO LONGER AVAILABLE

OBSOLETE FEATURE	REPLACE IN 5.2A BY
Infinite Element	**Infinite Element** is no longer defined in the physic interface. Infinite elements are defined in the Definitions section instead.
Electrochemical Heat flux	**Boundary Heat Source**
Reaction Heat Flux	**Boundary Heat Source**
Heat Transfer in Participating Media	**Heat Transfer in Solids** contributing to **Radiation in Participating Media**
Porous Matrix	**Heat Transfer in Porous Media**

Note that it is not possible to add any of the obsolete features from the COMSOL Desktop in version 5.2a. Only the new features are available.

OBSOLETE POSTPROCESSING VARIABLES DELETED IN 5.1

List of deleted postprocessing variables: `ndflux_acc`, `ndflux_acc_u`, `ndflux_acc_d`, `ntflux_acc`, `ntflux_acc_u`, `ntflux_acc_d`, `nteflux_acc`, `nteflux_acc_u`, `nteflux_acc_d`, `qout`, `qout_u`, `qout_d`, `qin`, `qin_u`, `qin_d`, `qw`, `qw_u`, and `qw_d`.

CHANGES IN VARIABLE NAMES

The variable `ht.Tvar`, used for nonisothermal flow wall functions, becomes a fully scoped variable in the following features:

- **Heat Flux** (`ht.hf1.Tvar`), **Line Heat flux**, **Point Heat flux**, and **Out of plane heat flux**.
- **Heat Source** (`ht.hs1.Tvar`), **Boundary heat source**, **Line heat source**, and **Point heat source**.
- **Diffuse Surface** (`ht.ds1.Tvar`).

The fully scoped variables `Ptot`, `Pbtot`, `Pltot`, and `Pstot` become `P0`, `Pb`, `Pl`, and `Ps`, respectively, in the following features:

- **Heat Flux**
- **Inflow Heat Flux**
- **Line Heat Flux**
- **Heat Source**
- **Boundary Heat Source**
- **Line Heat Source**
- **Line Heat Source** (**Thin Rod** subfeature)
- **Line Heat Source on Axis**
- **Layer Heat Source** (**Thin Layer** subfeature)

In the **Thermal Contact** feature:

- `Qfric`, `QfricPow`, and `Qrate` become `Qb`, `Pb`, and `Qbpart`, respectively
- `Qfric0`, `Qrate0`, `Econtact0`, `hrad0`, `hgap0`, and `hconstr0` are removed and migrated to existing variables `Qb`, `Qbpart`, `Econtact`, `hrad`, `hgap`, and `hconstr`, respectively, in old models.

The **Change Effective Thickness** feature from the Heat Transfer in Thin Shells interface merges `htsh.de` into the existing variable `htsh.ds`.

The Heat Transfer in Thin Shells interface also merges the property `de` and variable `htsh.de` into existing property `ds` and variable `htsh.ds`. An API command such as `model.physics("htsh").prop("de").set("de", [...])` now does not have any effect.

Backward Compatibility with Version 4.4

OUT-OF-PLANE HEAT TRANSFER UPDATE

The out-of-plane heat transfer property, which was available in 2D and 1D geometries, has been removed as of COMSOL Multiphysics 5.0. Now, all geometries are considered 3D geometries and thickness or area parameters are always available.

API Users

With the **Overall heat transfer rate** option (previously named **Total power** or **Total heat flux**), the thickness parameters (`dz_entr` and `Ac_input`) have been removed for the following features:

- **Heat Source**
- **Heat Flux**
- **Boundary Heat Source**
- **Layer Heat Source**
- **Inflow Heat Flux**

Compatibility with previous code is maintained for most of the cases. However, three configurations require manual updates of the code:

- If you specified the thickness parameter twice (there were two instances of specifying the parameter).

- If you first specified the thickness and in a second command the total power value.
- If you first specified the thickness and then activated the out-of-plane heat transfer option.

For these 3 cases, you need to rewrite this part of the Java® code by doing the following. For a heat source example (2D and 1D axisymmetric geometries): `Ptot_50 = Ptot_44_value*ht.d/dz_entr_44_value`,
where `Ptot_44_value` is the total power value used in previous versions, `Ptot_50` is the total power value used since COMSOL Multiphysics 5.0, and `dz_entr_44_value` is the value used in previous versions.

In 1D geometries, the following update should be performed: `Ptot_50 = Ptot_44_value*ht.d/Ac_input_44_value` ,
where `Ac_input_44_value` is the value used in previous versions.

Flux Variables

The flux variables are always given in W/m² (also for 2D or 1D geometries). To obtain the corresponding flux variables in W/m on a boundary in 2D, for example, the flux variables have to be multiplied by the thickness `ht.d`.

The following variables have been updated following this convention:

- Conductive heat flux, `ht.dflux`
- Convective heat flux , `ht.cflux`
- Translational heat flux, `ht.trlflux`
- Total energy flux, `ht.teflux`
- Total heat flux, `ht.tflux`
- Boundary convective heat flux, `ht.chflux`
- Radiative heat flux, `ht.rflux`

All flux variables now have the following sign convention: positive heat flux corresponds to heating, whereas negative heat flux corresponds to cooling. The variables `chflux` and `rflux` may have opposite signs compared to previous versions, depending on the context where they are defined.

INFLOW HEAT FLUX IS ONLY APPLICABLE ON NON-SOLID DOMAINS

The **Inflow Heat Flux** feature is no longer applicable on boundaries adjacent to the following domain features: **Heat Transfer in Solids**, **Biological Tissue**, and **Isothermal Domain**. This restriction avoids ill-posed configurations due to the null velocity field in these features. This may change model results if this boundary condition was previously applied to a boundary adjacent to a nonfluid domain.

HEAT TRANSFER WITH PHASE CHANGE NOW USES A VOLUMETRIC FORMULATION

The variables `ht.theta`*i* now represent volume fractions (they were mass fractions before). This implies changes in the definition of density and heat capacity at constant pressure close to the phase change interface. Far from the phase change interface, the density and heat capacity are unchanged.

Backward Compatibility with Version 4.3b

INTERIOR FLUXES UPDATED

Because the interior flux variable definitions have been revised, models that use interior fluxes in definitions, physics interfaces, or postprocessing show different results in versions 5.0 and later than in earlier versions.

IMPROVED ENERGY BALANCE

Changes introduced in the equation formulation to enable better energy balance may slightly modify the results.

Among the changes, the reference enthalpy, H_{Ref}, the value of which is defined up to a constant, is now set to 0 J/kg at the reference temperature and pressure. This changes the absolute values of the enthalpy and several variables based on it. In particular, the value of the convective heat flux is changed. Even the sign may be changed in some cases. However, the quantity of interest, the net convective heat flux, is identical because the shift induced by the change of H_{Ref} applies for the enthalpy definition at the inlet as well as at the outlet.

Backward Compatibility with Version 4.3a

HEAT TRANSFER IN POROUS MEDIA

Old models using the **Porous** feature will be supported in the COMSOL Multiphysics interface. However, it is recommended to replace this feature with Heat Transfer in Porous Media (see Obsolete Features). For a model using the API, it will be necessary to update the model to use Heat Transfer in Porous Media.

SURFACE-TO-SURFACE RADIATION

The surface radiosity, previously named `J`, is now named `ht.J` by default (assuming that `ht` is the physics interface tag).

Models created using versions 4.3 and 4.3a that contain surface-to-surface radiation support the old syntax in multiphysics interfaces. It is, however, recommended that you update the models to use the new syntax. Models created in a version before 4.2a need to be opened in version 4.3 or 4.3a and resaved before being opened in 5.0 or a later version.

For a model using the API, you must update the model to use the new syntax.

RADIATION IN PARTICIPATING MEDIA

The radiative intensity variable names, previously named `I1` and so on, became `ht.I1` (when there are fewer than 10 discrete directions) or `ht.I01` (when there are 10 or more discrete directions) by default.

Models created in versions 4.3 and 4.3a that include radiation in participating media support the old syntax in multiphysics interfaces. It is, however, recommended that you update the models to use the new syntax. Models created in a version before 4.2a need to be opened in version 4.3 or 4.3a and resaved before being reopened in 5.0 or a later version. In addition, for all versions, it may be necessary to regenerate the default solver to get the correct solver settings.

The S2 quadrature in 2D has been replaced by the LSE symmetric quadrature. It has been updated so that it coincides with the other quadratures. LSE symmetric quadratures were already used for S4, S6, and S8 in 2D and for all 3D quadratures. This quadrature update can result in significant changes in 2D models using the S2 quadrature. However, the theoretical error is not different and the 2D and 3D implementations are now consistent.

Backward Compatibility with Version 4.3

THE DEFAULT MODEL LIST HAS BEEN REMOVED

Heat transfer physics interfaces no longer have the default model list. This list was used to change the default feature for heat transfer models. You can obtain a similar result by adding the corresponding domain feature immediately below the default feature and setting its domain selection to **All domains**.

Any model files for Java® that modify the default model value require a manual update.

IMPROVED STABILIZATION OF HEAT TRANSFER IN SOLIDS

The streamline diffusion stabilization for **Heat Transfer in Solids** and **Biological Tissue** features has been improved. It now accounts for contributions from linear source terms from the **Heat Source**, **Out-of-Plane Convective Cooling**, **Out-of-Plane Radiation**, and **Out-of-Plane Heat Flux** features. This improves the robustness of the convergence when these contributions are large. This change may modify the convergence behavior of existing models.

FRAME SELECTION IN DISCRETIZATION SECTION

The **Frame type** list, which was previously available when a moving mesh was detected, has been removed. The frame type is now controlled by the features.

UPDATE OF FEATURE VARIABLE NAMES

In order to avoid name conflicts between feature variable names, some of them have been renamed. This change concerns contributive features. Because they are contributing, it is possible to have similar features active on the same boundary. To make it possible to distinguish the variables from each feature, the feature scope has been introduced in the variable name so that variable names now use a prefix. These changes affect the following feature variables:

- Heat flux
- Out-of-plane heat flux
- Convective cooling
- Out-of-plane convective cooling
- Boundary heat source
- Heat source
- Line heat source
- Point heat source
- Edge heat flux/Point heat flux
- Electrochemical reaction heat flux
- Reaction heat flux

For example, in a **Boundary Heat Source** feature, `ht.surf` has been renamed into `ht.bhs1.surf` (assuming that `ht` is the physics interface tag). In the Heat Flux feature, the variable that was previously named `ht.q0_hf1` is now `ht.hf1.q0`.

Any model files for Java® that use the old variable names in expressions (such as expressions used for plotting or data evaluation that include such old variable names) need to be updated manually.

NEW DEFAULT FLUID FEATURES AND OPAQUE SUBFEATURE

All MPH files created in version 4.3 are converted to appear and behave the same way as in 5.0 or a later version.

Due to these new default features, model files for Java can be simplified. In addition, model files for Java that add the **Fluid** feature or the **Opaque** subfeature with the default tag require a manual update to avoid duplicate tag conflicts.

WEAK CONSTRAINTS UPDATE FOR FLUID FLOW USER INTERFACES

The weak constraint formulations for some boundary conditions in the following interfaces have been updated:

- Laminar Flow
- Turbulent Flow, k-ε
- Turbulent Flow, Low Re k-ε
- Nonisothermal Flow
- Conjugate Heat Transfer

See the CFD Module release notes for details.

Weak constraints for the **Interior Wall** feature are no longer available.

REVISION TO THE TURBULENCE MODELS

The formulations of some variables in the turbulence models have been revised in order to improve accuracy. Simulations using a turbulence model can display a different convergence behavior in versions 5.0 and later compared to version 4.3, and the numerical result can differ slightly between the versions.

Obsolete Features

The following features have been removed:

- **Inflow Heat Flux** (replaced by **Inflow** and **Boundary Heat Source**)
- **Electromagnetic Heat Source** and **Electromagnetic Heat Source** (replaced by **Electromagnetic Heating**)
- **Boundary Thermoelectric Effect** (merged with **Thermoelectric Effect**)

- **Convective Heat Flux** (replaced by **Heat Flux** with **Convective heat flux option**)
- **Out-of-plane Convective Heat Flux** (replaced by **Out-of-plane Heat Flux** with **Convective heat flux option**)
- **Porous Matrix** (replaced by **Porous Medium**)
- **Infinite Element** (replaced by **Infinite Element Domain** declared in the **Definitions** node)
- **Electrochemical Heat Flux** (replaced by **Boundary Heat Source**)
- **Reaction Heat Flux** (replaced by **Boundary Heat Source**)

Heat Transfer in Participating Media (replaced by **Heat Transfer with Radiation in Participating Media**)

MEMS Module

New Functionality in Version 5.3a

MODIFIED PHYSICS INTERFACE: ELECTROMECHANICS

The Electromechanics physics interface has been upgraded to a predefined multiphysics interface, combining the Electrostatics and Solid Mechanics physics interfaces with the Electromechanical Forces multiphysics coupling and the Moving Mesh multiphysics definition to solve the structural equations together with the equations of electrostatics. The new approach improves the flexibility and design options for electromechanics modeling.

MODIFIED PHYSICS INTERFACE: THERMOELASTICITY

The Thermoelasticity physics interface has been upgraded to a predefined multiphysics interface, combining the Solid Mechanics and Heat Transfer in Solids physics interfaces with the Thermal Expansion and Temperature Coupling multiphysics couplings to solve for the displacement of the structure and the temperature deviations and resulting heat transfer induced by the thermoelastic coupling. The new approach improves flexibility and design options for thermoelasticity modeling.

MODIFIED PHYSICS INTERFACE: FLUID-STRUCTURE INTERACTION (FSI)

It is now possible for MEMS Module users with the CFD Module license to perform FSI simulation with two-phase or three-phase flow physics, thanks to the improved Moving Mesh machinery in the core functionality of COMSOL Multiphysics.

Backward Compatibility with Version 5.2

The friction slip velocity is no longer defined as a dependent variable. When running a model using the COMSOL API, you will need to remove the reference to this variable in the solver settings. The friction slip velocity is usually defined as `<comp>_<solid>_vslip_<pairname>`, where `<comp>` is the tag of the component, `<solid>` is the tag of the Solid Mechanics physics interface where the **Friction** node is defined, and `<pairname>` is the name of the contact pair.

Backward Compatibility with Version 4.3b

For models using contact, plot expressions are not updated when you open an old model where the pair names have been edited manually. Also, opening an old model where a pair name has been manually edited might affect the solver configuration. It is recommended that you regenerate a new default solver configuration.

Microfluidics Module

Backward Compatibility with Version 4.3a

The Free Molecular Flow and Transitional Flow user interfaces now form the Molecular Flow Module. A Molecular Flow Module license is required to use these interfaces.

Backward Compatibility with Version 4.3

BRINKMAN EQUATIONS AND FREE AND POROUS MEDIA FLOW

The momentum equations in the Brinkman Equations interface and the Free and Porous Media Flow interface have been corrected. The term $-Q_{\mathrm{br}}\cdot\mathbf{u}/\varepsilon_{\mathrm{p}}^2$ on the right-hand side previously lacked the factor $1/\varepsilon_{\mathrm{p}}^2$, where ε_{p} is the porosity.

MOBILITY

In physics interfaces modeling migration (for example, Transport of Diluted Species), anisotropic mobility is now supported, and the mobility can be set to follow the Nernst-Einstein relation. The default setting when making new models has been changed to the Nernst-Einstein relation. Model files for Java®-generated prior to 4.3b using migration will have to be modified manually to account for these changes.

Mixer Module

New Functionality in Version 5.3a

MORE FLEXIBLE ROTATING MACHINERY, FLUID FLOW INTERFACE

The functionality previously provided by separate Rotating Machinery, Fluid Flow interfaces is now embedded in the Single Phase Flow interfaces. The Rotating Machinery, Fluid Flow entries in the Model Wizard now add a Single Phase Flow interface and a Rotating Domain feature under Definitions, Moving Mesh in the Model Builder tree. This new design offers a more flexible modeling approach as well as new functionality for the modeling of rotating machinery equipment for fluid flow applications. In particular, all turbulence models supported by the Single Phase Flow interface can now be applied to rotating machinery simulations as well.

Molecular Flow Module

Backward Compatibility with Version 5.0 and Earlier

Due to the fact that multiple species are now supported, some variable names have changed. In general, variables will now have `_G` appended to their names, where `G` is the name specified for the **Incident molecular fluxes** in the **Dependent Variables** section. So, if a model is updated and resolved, the expressions used in results should be changed as follows:

- The number density should be changed from `fmf.N` to `fmf.N_G`.
- The pressure should be changed from `p` to `fmf.p_G`.
- The reconstructed number density should be changed from `fmf.N` to `fmf.ndr1.Nr_G`, where `ndr1` is the tag of the **Number Density Reconstruction** feature.
- Any other variables can be updated using the **Replace Expression** button available during results processing.

In addition, new variables exist for the total pressure, number density, and so on:

- The total number density, `fmf.ntot`.
- The total pressure, `fmf.ptot`.
- The total incident molecular flow, `fmf.Gtot`.

Multibody Dynamics Module

New Functionality in Version 5.3a

LUMPED MECHANICAL SYSTEM INTERFACE

A new **Lumped Mechanical System** physics interface has been added. In this physics interface, you can model displacements and forces in a mechanical system using a circuit representation with lumped components such as masses, springs, and dampers. In addition to the displacement and force sources in the system, terminal nodes are used to define forces, displacements, velocities, accelerations, and impedances as boundary conditions. The physics interface supports stationary, frequency-domain, eigenfrequency, and time-domain analysis. The models created with the Lumped Mechanical System interface can also include connections to distributed finite element models.

CAM-FOLLOWER

A new physics feature, **Cam-Follower**, has been added to the **Multibody Dynamics** interface. It is intended to force a point of one moving body to maintain contact with a curve or surface of another moving body. This connection node is generic in nature and allows any point to follow a curve or surface, including a typical radial or cylindrical cam-follower connection used in various mechanisms. The cam-follower connection models a simplified contact and establishes a higher kinematic pair between a cam and its follower.

COUPLING BETWEEN MULTIBODY DYNAMICS AND HYDRODYNAMIC BEARING

It is now possible to support shafts in a multibody system using various types of hydrodynamic bearings. The new multiphysics coupling feature **Solid-Bearing Coupling** is makes it possible to couple the **Multibody Dynamics** interface with the **Hydrodynamic Bearing** interface.

New Applications in Version 5.3a

LUMPED MODEL OF A VEHICLE SUSPENSION SYSTEM

In this example, a lumped model of a vehicle suspension system that has eleven degrees of freedom is analyzed. The **Mass**, **Spring**, and **Damper** nodes in the Lumped Mechanical

System interface are used to model the wheels, including the suspension system, as well as the seats with a passenger. The vehicle body, having three degrees of freedom, is modeled as a rigid body in the Multibody Dynamics interface.

The **External Source** node in the Lumped Mechanical System interface is used to connect the MBD model of the vehicle body to the lumped model of the rest of the system. A transient analysis is performed to compute the vehicle motion and the seat vibration levels for a given road profile.

LUMPED MODEL OF A HUMAN BODY

Several mass–spring–damper models have been developed to study the response of a human body. In such models, the lumped elements represent the mass of different body parts and stiffness and damping properties of various tissues.

In this example, a lumped model of a human body that has five degrees of freedom is analyzed. The model includes the shoe-ground interaction with the human body. The **Mass**, **Spring**, and **Damper** nodes in the Lumped Mechanical System interface are used to model the body, including the shoe and ground. First, an eigenfrequency study is performed to find out the natural frequencies of the system and then a frequency response analysis is performed to compute the system response for a specified base excitation.

LUMPED LOUDSPEAKER DRIVER USING LUMPED MECHANICAL SYSTEM

This is a model of a moving-coil loudspeaker where a lumped parameter analogy represents the behavior of the electrical and mechanical speaker components. This lumped model is coupled to a 2D axially symmetric pressure acoustics model describing the surrounding air domain.

This example illustrates an alternate way of modeling mechanical components (that is, mass, springs, and dampers) using the Lumped Mechanical System interface in a lumped loudspeaker driver model.

MODELING A RADIAL CAM-BASED VALVE OPENING MECHANISM

In this example, a spring-loaded valve opening mechanism that has a rocker arm and a radial cam is studied. All the components of the system are modeled as rigid, and are connected through prismatic, hinge, and slot joints. The cam-follower connection as well as other joint connections are modeled using built-in nodes of the Multibody Dynamics interface.

A transient analysis is performed for various values of the valve spring stiffness. The output from the model includes the follower velocity, follower acceleration, cam-follower connection force, and the torque required to rotate the cam shaft.

Nonlinear Structural Materials Module

New Functionality in Version 5.3a

ENHANCEMENTS TO POROUS PLASTICITY MODELS

A new option for modeling void growth for porous plasticity models has been added. The **Porous Plasticity** subnode available for **Linear Elastic Material** and **Nonlinear Elastic Material** includes the possibility to add void growth in shear and void nucleation in tension.

The porous plasticity material models have also been augmented with isotropic plastic hardening. These options are available for all porous plasticity models.

NEW SHAPE MEMORY ALLOY MATERIAL MODELS

Shape memory alloys are becoming increasingly popular, particularly for medical applications. The mechanical description of shape memory alloys is complex, since the behavior of these materials is caused by phase transformations triggered by either mechanical strains or changes in temperature.

Two of the most common mathematical models for describing shape memory alloys have been added:

- Souza-Auricchio
- Lagoudas

NAME CHANGE OF CREEP VARIABLE

In the Navarro-Herring creep model, the name of the parameter `solid.lemm1.cmm1.b` is changed to `solid.lemm1.cmm1.b_nav`.

New Applications in Version 5.3a

SHAPE MEMORY ALLOY MODEL

This tutorial model demonstrates the Shape Memory Alloy constitutive law under uniaxil loading. This model can be used as a benchmark for comparing analytical and lab tests.

Optimization Module

Backward Compatibility with Version 4.3a

The **Optimization** check box in the Stationary and Time Dependent study steps have been removed. Instead, use the Optimization study step. The **Optimization** check box does appear in models created in earlier versions of COMSOL Multiphysics where you have selected that check box.

Particle Tracing Module

New Functionality in Version 5.3a

NULL COLLISION METHOD FOR MONTE CARLO COLLISION MODELS

A new algorithm for detecting random collisions between model particles and a rarefied background gas is available. In the **Collisions** feature for the Charged Particle Tracing interface, select an option from the **Collision detection** list: **At time steps taken by solver** or **Null collision method, cold gas approximation**.

The **Null collision method, cold gas approximation** allows multiple collisions per particle to be detected within each time step taken by the solver. This allows accurate Monte Carlo collision modeling, even if the time steps taken by the solver are very large. This algorithm is only valid for highly energetic model particles because it neglects the thermal velocity of the background gas.

DISTRIBUTIONS OF PARTICLE RELEASE TIMES

When releasing particles, you can now select a **Distribution function** for the particle release times. Built-in options to specify a **List of values** or to sample times from a **Normal**, **Lognormal**, or **Uniform** distribution are available.

THERMAL RE-EMISSION BOUNDARY CONDITION

The new **Thermal Re-Emission** boundary condition causes incident particles to be adsorbed at a wall and then emitted with a randomly generated velocity based on the wall temperature. The distribution of release angles follows the cosine law. The same behavior can be modeled at an **Inlet** by selecting the **Thermal** velocity distribution.

REUSE PARTICLES FOR SECONDARY EMISSION

The **Reuse particle degrees of freedom** list is found in the **Advanced Settings** section of the settings window for the particle tracing interfaces. Using the options in this list, you can recycle degrees of freedom by rereleasing a particle that had disappeared at an earlier time. In principle, this can be used to release a very large number of particles over time with significantly fewer degrees of freedom.

CYLINDRICAL AND HEXAPOLAR GRID RELEASE

The **Release from Grid** feature can now release particles in a **Cylindrical** or **Hexapolar** grid in 3D. The **Cylindrical** grid release samples the particle positions at all combinations of the given radial and azimuthal coordinates. The **Hexapolar** grid release produces a number of concentric rings of particles, with each ring containing six more particles than the one preceding it.

IMPROVED PERIODIC ELECTRIC AND MAGNETIC FORCES

The **Electric Force** and **Magnetic Force** can now exert electromagnetic forces that are periodic but not time harmonic. Select an option from the **Time dependence of field** list: **Stationary or time dependent**, **Time harmonic**, or **Periodic**. With the **Periodic** option, it is possible to solve for the field over one period and then extend the applied force to an arbitrary number of periods, even if the field is not solved for in the frequency domain.

New Applications in Version 5.3a

DISPERSION OF HEAVY PARTICLES IN A TURBULENT CHANNEL FLOW

In this benchmark model, solid particles are released in a fully developed turbulent channel flow. The particles are subjected to a drag force that includes contributions from the fluid turbulence, implemented using a Continuous Random Walk (CRW) model. Because the turbulence in the channel is anisotropic, particles with significant inertia tend to cluster near the channel walls, while very small particles remain uniformly distributed throughout the channel cross section.

Backward Compatibility with Version 5.3 and Earlier

NOMENCLATURE CHANGES FOR MULTIPLE PARTICLE SPECIES

The **Override Properties** feature has been renamed to **Particle Properties** to emphasize that it can be used to release multiple species of particles in the same model. The list **Inherit Properties** in the settings for most particle release features, such as **Inlet** and **Release from Grid**, has been renamed to **Released particle properties**.

TURBULENT DISPERSION CHANGES

The turbulent dispersion models used by the **Drag Force** in the Particle Tracing for Fluid Flow interface have been modified and a deprecated turbulent dispersion model has been removed.

The option **Discrete random walk, variable time step** has been removed from the **Turbulent dispersion model** list. When an old model using this obsolete option is loaded in version 5.3a, the **Discrete random walk** option is now selected. However, **Discrete random walk** actually behaves more like the old **Discrete random walk, variable time step** model because it only samples the turbulent velocity perturbation at discrete time intervals based on the eddy lifetime or eddy crossing time in the flow.

CHANGES TO RELEASE FROM EDGE AND POINT

In the Particle Tracing for Fluid Flow interface, the **Release from Edge** and **Release from Point** features now allow the initial particle velocity to be selected from a list containing previously computed velocity fields, similar to the behavior of the **Release** and **Inlet** features. This makes the **Release from Edge** and **Release from Point** features easier to couple with another physics interface that models the fluid flow.

CHANGES TO PERIODIC ELECTRIC AND MAGNETIC FORCES

In models built in version 5.3 or earlier, if an **Electric Force** or **Magnetic Force** was present and the **Multiply force by phase angle** check box was selected, then in version 5.3a, **Time harmonic** will be selected from the new **Time dependence of field list**, which replaces this check box.

Backward Compatibility with Version 5.2a and Earlier

REFERENCE EDGE FEATURES HAVE BEEN REMOVED

The **Reference Edge** subnode, which was previously used to specify the orientation of the transverse beam directions for the **Particle Beam** feature in 3D, has been removed. Instead, the orientation of the transverse beam directions is specified using an extra selection field in the settings window for the **Particle Beam** feature itself. Similarly, the **Reference Point** subnode for specifying the beam center has been replaced by an extra selection in the settings window for the **Particle Beam** feature.

Backward Compatibility with Version 5.2

TURBULENT DISPERSION CHANGES

In the settings window for the **Drag Force** for the Particle Tracing for Fluid Flow interface, the **Turbulent dispersion** check box has been replaced by the **Turbulent dispersion model** list. If the check box is selected in a model created in version 5.2 or earlier, then after opening the model in version 5.2a, the option **Discrete random walk** will be selected from the list. If the check box is cleared, then **None** will be selected from the list.

PARTICLE DENSITY SPECIFICATIONS MERGED

In version 5.2, when specifying particle properties using the **Particle Properties** node or **Override Properties** node in the Particle Tracing for Fluid Flow interface, it is possible to specify two different values of the particle density, depending on whether the **Compute particle mass** check box is selected. In version 5.2a, these two values have been merged into a single parameter value; that is, the value or expression for the **Particle density** will be retained when selecting or clearing the **Compute particle mass** check box in the physics interface **Advanced Settings** section.

PARTICLE BEAM CHANGES

The names of several settings have been changed in the **Particle Beam** feature for the Charged Particle Tracing interface.

In the **Sampling from phase space distribution** list, **Uniform** has been renamed **KV**.

In the **Longitudinal velocity distribution** list, **Uniform** has been renamed **None** and **Gaussian** has been renamed **Normal**.

The parameters **Position refinement factor** and **Release distribution accuracy order** have been removed. Due to some changes in the algorithm for initializing particle positions, in version 5.2a, the distribution of particle positions no longer depends on the finite element mesh and may differ from the distribution in version 5.2.

Backward Compatibility with Version 5.1

RENAMED PHYSICS INTERFACE SETTINGS

The **Release type** list in the settings windows for the Charged Particle Tracing and Particle Tracing for Fluid Flow interfaces in earlier versions has been renamed **Particle release specification**. For the Charged Particle Tracing interface, the options **Static** and **Transient** have been renamed to **Specify current** and **Specify release times** (the default), respectively. For the Particle Tracing for Fluid Flow interface, the options **Static** and **Transient** have been renamed to **Specify mass flow rate** and **Specify release times** (the default), respectively.

Backward Compatibility with Version 5.0 and Earlier

OLD NODES FOR PARTICLE-FIELD AND FLUID-PARTICLE INTERACTIONS ARE OBSOLETE

The old **Particle-Field Interaction** and **Fluid-Particle Interaction** nodes, which could be added directly to the Charged Particle Tracing and Particle Tracing for Fluid Flow interfaces, respectively, are now obsolete. They will be removed in a future version. Simulations of particle-field and fluid-particle interactions should instead use the dedicated multiphysics nodes.

ELASTIC COLLISION FORCE NODE HAS BEEN REPLACED

If a model created in version 5.0 or earlier containing the **Elastic Collision Force** node is opened in version 5.1 or later, this node will be replaced. If the **Collision model** in the old **Elastic Collision Force** node was set to **Monte Carlo**, a **Collisions** node with the **Elastic** subnode will be created. If instead the **Collision model** was set to **Friction**, the **Friction Force** subnode will be added.

Pipe Flow Module

Backward Compatibility with Version 5.1 and older

REVISED FORMULATION OF THE PIPE FLOW INTERFACE

The Pipe Flow interface has been amended in version 5.2 with an inertial term. The stabilization and element shape functions have also been changed. Models created in previous versions will be opened using the old interface with a notification. For new models, only the new interface is available in the physics interface list.

Plasma Module

New Functionality in Version 5.3a

NEW PHYSICS INTERFACE

A new physics interface, **Plasma, Time Periodic**, has been added for modeling capacitively coupled plasmas (CCP). Instead of solving the problem in the time domain, the periodic steady state solution is computed. This avoids having to solve for tens or hundreds of thousands of RF cycles, which is typically how long it takes before the plasma reaches the periodic steady state solution. This novel approach maintains all the nonlinearity of the model while dramatically reducing computation time. The physics interface accomplishes this by attaching an extra dimension to the underlying mathematical equations representing one RF cycle, and enforcing periodic boundary conditions in the aforementioned extra dimension. This means that 1D models take seconds to solve, and 2D models typically take on the order of an hour for a given power input.

New and Updated Applications and Models in Version 5.3a

NEW MODELS

- Alpha to Gamma Transition
- GEC CCP Reactor, 2D
- CCP Ion Energy Distribution Function
- Computing the Plasma Impedance
- Impedance Matching

Backward Compatibility with Version 5.3 and Earlier

There have been several changes that could affect the solution obtained when running a model in version 5.3a versus 5.3 or earlier. This is due to the following changes and bug fixes:

- When electron impact reaction rate constants are computed from cross section data, due to a bug, the result will be different for the Maxwellian or Druyvesteyn electron energy distribution functions. For DC discharges, where Townsend coefficients are recommended, this should not make much of a difference. For inductively coupled or microwave discharges, the average electron density and plasma potential could change by as much as 5%, but typically around 2%. If significant differences in the solution are obtained, it is recommended that you contact `support@comsol.com`.

- Surface species are no longer available from the context menu. Instead, surface species will appear in the Model Builder when a surface reaction containing surface species is added. If the surface reaction is deleted, then the dependent surface species will also be deleted.
- The rate constant for a surface reaction that uses sticking coefficients has changed, and is now proportional to the molecular weight of the reactant instead of the mean molecular weight. In systems where the molecular weight is the same for all species, this change won't affect the results. In models containing species with very different molecular weights, the surface reaction rates may change according to the square root of the molecular weight of the reactant divided by the mean molecular weight. In such systems, it is recommended to activate the new **Mixture diffusion correction** property, which adds additional correction terms to the mass flux vector for the heavy species. Overall discharge properties should not be significantly affected by this change.
- The definition of surface site concentrations and surface reaction rate rates has been corrected for models that include a surface chemical mechanism where the surface species have different site occupancy numbers. This should only result in small changes to the solution, if any. The total surface site concentration is now a property on the physics interface level, rather than computed for each species on the feature level.
- The equation to compute binary diffusion coefficients from characteristic length parameters has changed. This may produce very slight changes in the diffusion coefficients for the heavy species in the Plasma and Heavy Species Transport interfaces. Any change in the solution due to this should be a tiny fraction of a percent.

Backward Compatibility with Version 5.2 and Earlier

The old Inductively Coupled Plasma and Microwave Plasma interfaces can still be opened, edited, and solved. When such a model is opened, a warning is given indicating that the interface is obsolete and will eventually be removed.

Backward Compatibility with Version 5.1

Models using the Reduced Electric Fields study will interpret the unit in the edit field for the Reduced electric fields as Townsend. For example, when opening a model created in version 5.1, if the value for the Reduced electric fields was 10, this will now appear as `10[Td]` in the study settings.

Backward Compatibility with Version 4.4

Old models will be updated to use the new boundary conditions, but re-solving a model still gives the same results as before. The following changes to the model are made upon opening:

- If an **Electric Potential** feature is applied to boundaries adjacent to a **Plasma Model** feature, it is converted to a **Metal Contact** feature. The settings from the original model are retained. If an **Electric Potential** feature is applied to boundaries adjacent to a **Charge Conservation** feature, it remains the same.
- If an **Electric Potential** feature is applied to a combination of boundaries adjacent to a **Plasma Model** and **Charge Conservation** feature, a **Metal Contact** feature is created for the boundaries adjacent to the **Plasma Model** feature, and the **Electric Potential** feature remains on boundaries adjacent to the **Charge Conservation** feature.
- Any **External Surface Charge Accumulation** boundary conditions are replaced by a new **Dielectric Contact** feature. The settings from the original model are retained. The **External Surface Charge Accumulation** boundary condition is now obsolete.

Ray Optics Module

New Functionality in Version 5.3a

CYLINDRICAL AND HEXAPOLAR GRID RELEASE

The **Release from Grid** feature can now release rays in a **Cylindrical** or **Hexapolar** grid in 3D. The **Cylindrical** grid release samples the ray positions at all combinations of the given radial and azimuthal coordinates. The **Hexapolar** grid release produces a number of concentric rings of rays, each ring containing six more rays than the one preceding it.

AUTOMATIC STOP CONDITION BASED ON REFLECTION COUNTER

A built-in option to **Count reflections** is available for the Geometrical Optics interface. When this option is enabled, a built-in stop condition in the **Ray Tracing** study step can be used to terminate the study when all rays have been reflected by at least the specified number of times.

MIRROR BOUNDARY CONDITION

A specialized **Mirror** boundary condition is now available in the Geometrical Optics interface. The **Mirror** boundary condition always specularly reflects incident rays.

CONTROL THE RELEASE OF REFLECTED RAYS AT MATERIAL DISCONTINUITIES

The **Material Discontinuity** feature now includes a list called **Release reflected rays**, with options **Always**, **Never**, or **Based on logical expression**. Use this option to more precisely control which boundaries in the model are able to release reflected rays when the incident ray is refracted. This option has no effect if the incident ray undergoes total internal reflection.

NEW PARTS FOR THE PART LIBRARY

The Part Library for the Ray Optics Module has been expanded to include the following:

- Asymmetric Spherical Lens: specify the curvature of each surface of the lens separately.
- Circular Planar Annulus: An annular ring that can be used to define apertures in a camera or telescope.

- On Axis Conic Mirror: An axisymmetric conic mirror with an optional center hole.
- Off Axis Conic Mirror: A conic mirror where the center is displaced from the symmetry axis.

New Applications in Version 5.3a

DOUBLE GAUSS LENS

This tutorial shows how to set up a multielement objective lens consisting of six asymmetric spherical lenses. The results of a grid-based ray trace at a single wavelength and field angle are shown graphically.

DOUBLE GAUSS LENS (PARAMETRIC SWEEP)

This example is an extension of the *Double Gauss Lens* example described earlier. It includes a **Parametric Sweep** over different wavelengths and field angles.

PETZVAL LENS

A Petzval lens consists of two positive lens groups surrounded by air. In the field-flattened Petzval lens, a negative lens element is also placed as close as possible to the focal plane. The results of a grid-based ray trace at a single wavelength and field angle are shown graphically.

WHITE PUPIL ÉCHELLE SPECTROGRAPH

Échelle spectrograms are commonly used in astronomy for high-resolution analyses of stellar atmospheres and for precise Doppler velocimetry. This tutorial simulates a “white pupil” form of this instrument. The model geometry includes two mirrors, a Petzval lens system, and two grating components: an Échelle grating and a cross disperser. Together, these two gratings, with their orthogonal directions of periodicity, produce a 2D array of spots called an Échelle diagram.

Backward Compatibility with Version 5.3 and Earlier

DIFFRACTION GRATING SETTINGS MOVED TO A SUBNODE

In version 5.3 and earlier, the **Grating** feature for the Geometrical Optics interface automatically released transmitted and reflected rays of diffraction order zero. In version 5.3a, the rays of diffraction order zero are controlled by a **Diffraction Order** subnode, as are all other diffraction orders that might be released.

NEW OPTIONS FOR SPECIFYING DIFFRACTION GRATING ORIENTATION

In the **Grating** feature for the Geometrical Optics interface, the options to control grating orientation in 3D have been reorganized and expanded. It is now possible to specify either the direction of the grating lines or the direction of periodicity in the grating surface. The input **Grating orientation specification** has been renamed **Direction of grating lines**.

NEW OPTIONS FOR SPECIFYING REFLECTANCE OR TRANSMITTANCE

In the Geometrical Optics interface, the settings to control **Thin dielectric films on boundary** in the settings for the **Material Discontinuity** feature have been renamed and rearranged.

The old options **Single layer, specified reflectance** and **Single layer, specified transmittance** have been renamed to **Specify reflectance** and **Specify transmittance**, respectively. For both of these options and for the **Anti-reflective coating** option, which has not been renamed, the **Treat as single layer dielectric film** check box is selected when opening the model in version 5.3a. Otherwise, this check box is cleared, which is the default behavior.

ADJUSTMENT TO THE RAY TERMINATION FEATURE

In the **Ray Termination** feature for the Geometrical Optics interface, the **Termination criteria** list has been renamed **Additional termination criteria**.

REMOVED CHECK BOXES FOR OBSOLETE FUNCTIONALITY

In the settings for the Geometrical Optics interface, the **Allow propagation outside selected domains** check box has been removed. This check box was flagged as obsolete in version 5.3, so clearing it would have created a Warning node.

In the Settings window for the **Release from Grid** feature, the check box **Suppress interaction with coinciding exterior boundaries** has also been removed.

Backward Compatibility with Version 5.2a and Earlier

REFERENCE EDGE FEATURES HAVE BEEN REMOVED

The **Reference Edge** subnode, which was previously used to specify the orientation of some optical components in 3D, has been removed. Instead, the orientation of these components is specified directly in the parent feature settings using an extra selection input. The following features are affected:

- **Linear Polarizer**

- **Linear Wave Retarder**
- **Mueller Matrix**
- **Grating**

Backward Compatibility with Version 5.2

ANISOTROPIC REFRACTIVE INDICES CAN NO LONGER BE SPECIFIED

In version 5.2 and earlier, it is possible to specify an anisotropic refractive index in the settings window for the **Medium Properties** node, despite the fact that the anisotropy of the refractive index is not taken into account when modeling ray propagation. In version 5.2a, only isotropic refractive indices can be specified in the settings window for the **Medium Properties** node. If an anisotropic refractive index was specified in an older version, then when opening the model in version 5.2a, all of the diagonal terms will be populated using the first diagonal value and all off-diagonal terms will be set to zero.

RENAMED INTENSITY COMPUTATION METHODS

The options in the **Intensity Computation** list in the settings window for the Geometrical Optics interface have been renamed.

- **Using principal curvatures** has been renamed to **Compute intensity**.
- **Using principal curvatures and ray power** has been renamed to **Compute intensity and power**.
- **Using curvature tensor** has been renamed to **Compute intensity in graded media**.
- **Using curvature tensor and ray power** has been renamed to **Compute intensity and power in graded media**.

RENAMED POLARIZATION SETTING

In the **Initial polarization type** list, available for most ray release features in the Geometrical Optics interface, the option **Partially coherent** has been renamed to **Partially polarized**.

CHANGES TO PERTURBATIONS FOR SOLAR RADIATION

The treatment of perturbations due to sunshape and surface roughness has been changed in version 5.2a. This change affects the **Illuminated Surface** and **Solar Radiation** features for the Geometrical Optics interface.

The **Use corrections for finite source diameter** check box has been replaced by the **Corrections for finite source diameter** list. If the check box was selected, **Create light cones at release points** will be selected from the list after opening the model in version 5.2a. If the check box was cleared, **None** will be selected from the list.

For the **Illuminated Surface**, the **Standard deviation of the angle of incidence** parameter has been renamed **Surface slope error** and is now used to define a Rayleigh distribution for perturbations to the surface normal, rather than the angle of incidence of radiation. To account for the different effect that the surface slope error has on the model, any value or expression entered for the **Standard deviation of the angle of incidence** in version 5.2 or earlier will be divided by 2, before being used to specify the **Surface slope error** in version 5.2a.

The **Number of rays in wave vector** space $N_{w,r}$ has been removed. The effect of surface roughness can no longer multiply the number of degrees of freedom created by a release feature at each release point.

ACTIVATION CONDITIONS ON OPTICAL COMPONENTS

The following boundary conditions require some form of ray intensity computation in order to have any effect on the solution:

- **Linear Polarizer**
- **Linear Wave Retarder**
- **Circular Wave Retarder**
- **Ideal Depolarizer**
- **Mueller Matrix**

If any of these features is present in a model created in version 5.2 or earlier, and **None** is selected from the **Intensity computation** list in the settings window for the Geometrical Optics interface, then the feature will be automatically disabled when the model is opened in version 5.2a. To enable these features, select any option other than **None** from the **Intensity Computation** list.

Backward Compatibility with Version 5.0

RENAMED SETTINGS

The Wall condition **Bounce** has been renamed to **Specular reflection**.

CHANGES TO FREQUENCY-DEPENDENT MATERIAL PROPERTIES

The **Frequency-dependent refractive indices** check box has been renamed to **Allow frequency distributions at release features**.

Expressions involving the ray frequency must always be included within the `noenv()` operator. Some expressions for user-defined refractive indices may fail to be evaluated properly in version 5.2a unless this operator is applied.

THIN DIELECTRIC FILM SETTINGS

In version 5.0, if the ray intensity is computed, the option to add **Thin Dielectric Film** nodes to a **Material Discontinuity** node is always available. In version 5.2a, the Thin Dielectric Film subnode can only be added if **Add layers to surface** or **Add layers to surface, repeating** is selected from the **Thin dielectric films on boundary** list in the **Material Discontinuity** settings window. If a model created in version 5.0 is opened in version 5.2a, and any **Thin Dielectric Film** nodes have been added, then **Add layers to surface** is selected from the **Thin dielectric films on boundary** list instead of the default **None**.

CHANGES TO THE CIRCULAR WAVE RETARDER

The **Retardance** of the **Circular Wave Retarder** now indicates the rotation angle of the polarization ellipse of a ray that passes through the device. In version 5.0, the retardance corresponds to half of this rotation angle. The behavior of the **Circular Wave Retarder** is now consistent with the documentation. If a model created in version 5.0 is opened in version 5.2a, the expression for the **Retardance** of any **Circular Wave Retarder** node is multiplied by 2.

OBSOLETE DEPOSITED RAY POWER NODE

The **Deposited Ray Power** node for absorbing domains is obsolete and will be removed in future versions. For simulations in which rays generate heat due to absorption, the **Ray Heat Source** multiphysics node should be used instead.

RF Module

New Functionality in Version 5.3a

RF MATERIAL LIBRARY

The RF Module has introduced a Material Library consisting of a number of substrate material data that assist in modeling RF components.

ADAPTIVE FREQUENCY SWEEP

The **Adaptive Frequency Sweep** study step is added to compute the response of a linear or linearized model subjected to harmonic excitation for several frequencies with a fine frequency resolution using a reduced-order model in the frequency domain. The Asymptotic Waveform Evaluation (AWE) model reduction is performed by a moment matching technique, where Padé approximation or a Taylor series expansion is used for the transfer function in a specified frequency interval.

RF PART LIBRARY

The RF Part Library now includes the edge launch connectors from Signal Microwave that assist in modeling RF components supporting high-speed connection and high data rates.

ADDITIONAL IMPROVEMENTS

- The phase of the de-embedded S-parameters can be adjusted from the calculated S-parameters with the propagation constant and the value of `doffset`. The de-embedding functionality is triggered when `doffset` is set to a nonzero value, and the S-parameter phase is scaled by the user-specified offset. It is assumed that the domain between the port boundary and the boundary projected by the `doffset` is straight, while maintaining a constant cross-sectional shape.
- A new arrow visualization based on the power flow direction is available. It is now possible to switch the direction of both visualization and power flow.
- The physics-controlled mesh can analyze the material properties characterized by an interpolation function with a frequency input argument and generate the appropriate size of mesh.
- In the **Time Dependent with FFT** preset study, the **Combine Solutions** study extension step filters out the unwanted parts of the results. For instance, the portion of

frequency spectrum after time-to-frequency FFT, such as the first and last 5% of the frequency spectrum, can be removed using a user-defined expression.

- Porous media has been renamed as effective medium.

New and Updated Models and Applications in Version 5.3a

NEW MODEL

The new Wide Band RCS Calculation Using Time-to-Frequency FFT model shows how to calculate the radar cross section (RCS) of a scatterer in a wide frequency range with the Electromagnetic Waves, Time Explicit physics interface. The problem is solved in the scattered-field formulation, where the background field is a temporally modulated Gaussian pulse. The simulated results present the scattered field in the frequency domain and time domain and the RCS per unit length of a circle in the frequency domain.

UPDATED MODELS

The application library examples are polished to present the postprocessing features introduced recently. These will enhance the postprocessing visualization and results analysis process.

UPDATED TUTORIALS USING REDUCED-ORDER MODELING TECHNIQUES

Using asymptotic waveform evaluation:

- *Evanescent Mode Cylindrical Cavity Filter*
- *RF Coil*

Using frequency-domain modal analysis:

- *Cascaded Rectangular Cavity Filter*
- *Coupled Line Filter*
- *CPW Bandpass Filter*
- *Waveguide Iris Filter*

Backward Compatibility with Version 4.3a

The following models have been moved from the RF Module Application Library to the Wave Optics Module Application Library:

- *Beam Splitter*

- *Photonic Crystal*
- *Scattering on Substrate*
- *Optical Scattering Off of a Gold Nanosphere*
- *Dielectric Slab Waveguide*
- *Fabry-Perot Cavity*

Rotordynamics Module

New Functionality in Version 5.3a

ROLLING ELEMENT BEARING

Models for six types of rolling element bearings have been added to the Rotordynamics Module:

- Deep groove ball bearing
- Angular contact ball bearing
- Self-aligning ball bearing
- Spherical roller bearing
- Cylindrical roller bearing
- Tapered roller bearing

Each of the bearings can have either a single row or a double row of rolling elements.

The bearing model includes a nonlinear representation of the contact stiffness between the rolling elements and the inner and outer races.

You can add the **Radial Roller Bearing** to both the Solid Rotor and Beam Rotor interfaces.

HYDRODYNAMIC THRUST BEARING

The Hydrodynamic Bearing interface can now solve for a **Hydrodynamic Thrust Bearing**. The bearing can be of the types **Tilted Pad**, **Tapered**, or **User Defined**. The tilted pad bearing can have either a point pivot or a line pivot. Cavitation of the lubricant can be included in the formulation.

MULTIPHYSICS COUPLING BETWEEN BEARINGS AND SOLIDS

A new multiphysics coupling, **Solid-Bearing Coupling**, has been added. With this coupling, you can connect a journal bearing modeled in the Hydrodynamic Bearing interface to a solid, either in the Solid Mechanics or Multibody Dynamics interfaces.

DEFAULT CAMPBELL PLOT

When an eigenfrequency analysis is performed in the Solid Rotor or Beam Rotor interfaces, default Campbell plots are now generated. For the Solid Rotor interface, Campbell plots are created both for the corotating frame and the fixed frame.

STABILITY OF A TURBOCHARGER UNDER THE INFLUENCE OF CROSS-COUPLED BEARING FORCES

Cross-coupled forces present in a hydrodynamic bearing often act as a negative damping in a rotor. Near the critical speed, this may lead to uncontrolled vibration of the turbocharger causing a risk of bearing failure. In this example, the influence of the cross-coupling forces on the dynamics of the rotor is analyzed. How to reduce this vibration is also studied.

EFFECT OF ROLLER BEARING CLEARANCE ON NONSYNCHRONOUS VIBRATION OF A ROTOR

Bearing clearances should be kept to a minimum to avoid nonsynchronous vibration of a rotor. However, a tight clearance reduces the durability of the bearing. This example illustrates modeling of the vibration induced by nonlinear contact for different radial clearances.

Semiconductor Module

New Functionality in Version 5.3a

SEMICONDUCTOR EQUILIBRIUM STUDY

A new **Semiconductor Equilibrium** study step is introduced for the **Semiconductor** physics interface. It is suitable for systems known to be in equilibrium, as well as for generating initial conditions for nonequilibrium systems.

QUASI-FERMI LEVEL FORMULATION

This is a new discretization scheme using the quasi-Fermi levels as the dependent variables for the charge carriers. The quasi-Fermi level formulation provides an alternative option for tackling the often highly nonlinear equation system when modeling semiconductor devices, such as at very low temperatures.

POWER-DRIVEN TERMINALS

A new option is added to the Metal Contact boundary condition. In addition to voltage-driven, current-driven, and connecting to circuits in two different ways, a power-driven option is added.

TRAPPING FUNCTIONALITY

The functionality of the Trapping feature is expanded. You can enter the trap occupancy and the degeneracy factor individually for each discrete or continuous energy level subnode. The energy discretization (energy range and number of mesh points along the energy axis) can also be tailored individually for each continuous energy level subnode.

PMLS FOR THE SCHRÖDINGER EQUATION

In addition to the Open Boundary condition for outgoing waves, the Perfect Matched Layer (PML) functionality is added to the Schrödinger Equation interface to absorb outgoing waves for stationary studies.

CORRECTIONS AND BACKWARD COMPATIBILITY

- The formulation for band gap narrowing (FVM and FEM), position-dependent band gap (FVM), and carrier diffusion due to temperature gradients (FVM) has been fixed. The MOSFET series of tutorial models, which use the band gap narrowing feature, is slightly affected.

- The Fermi-Dirac variable definitions for the finite element log formulation has been corrected.
- Fixed formulation for the case of a user-defined Schottky barrier height in the Metal Contact boundary condition.

New Models in Version 5.3a

GROSS–PITAEVSKII EQUATION FOR BOSE–EINSTEIN CONDENSATION

This tutorial model solves the Gross–Pitaevskii equation for the ground state of a Bose–Einstein condensate in a harmonic trap, using the Schrödinger Equation physics interface in the Semiconductor Module. The equation is essentially a nonlinear single-particle Schrödinger equation, with a potential energy contribution proportional to the local particle density. The eigenvalue study is not suitable for solving this kind of nonlinear eigenvalue problem. Instead, a stationary study is used with a global equation enforcing the normalization of the wave function to solve for the ground state solution. The result for a large number of particles compares well with the Thomas–Fermi approximation as expected.

SMALL-SIGNAL ANALYSIS OF A MOSCAP

The metal-silicon-oxide (MOS) structure is the fundamental building block for many silicon planar devices. Its capacitance measurements provide a wealth of insight into the working principles of such devices. This tutorial constructs a simple 1D model of a MOS capacitor (MOSCAP). Both the low- and high-frequency C-V curves are computed using the approach of small-signal analysis. The model employs the quasi-Fermi level formulation and the Semiconductor Equilibrium study step, both new in version 5.3a.

Structural Mechanics Module

New Functionality in Version 5.3a

NEW FLUID-STRUCTURE INTERACTION INTERFACE

A new Fluid-Structure Interaction interface has replaced the one used in previous versions. The new interface is of the modern style, based on a multiphysics coupling. When you add **Fluid-Structure Interaction** from the Model Wizard, the following features are added:

- A Solid Mechanics interface.
- A Laminar Flow interface, which can be modified to Turbulent Flow.
- A **Moving Mesh** node under **Definitions**, which makes it possible to use moving mesh in the fluid domains.
- A **Fluid-Structure Interaction** multiphysics coupling.

With the new approach, all functionality in the constituent physics interfaces is available.

In models created in program versions prior to 5.3a, the old Fluid-Structure Interaction interface will still be used.

GENERALIZED PLANE STRAIN

For 2D solid mechanics, a generalized plane strain formulation has been developed in addition to the existing plane strain and plane stress specializations. It is intended for modeling the central part of structures that are long with a constant cross section. As opposed to a standard plane strain formulation, nonzero out-of-plane strains are present.

BOLT THREAD CONTACT MODELING

When modeling bolted connections, it is seldom possible to include the actual geometry of the threads due to model size considerations. Still, the stress state in the vicinity of the bolt hole can be significantly influenced by the wedging effect caused by the contact pressure between the internal and external thread. The new **Bolt Thread Contact** feature in the Solid Mechanics interface makes it possible to model the two threaded parts using only cylindrical surfaces, and still incorporate the salient effects of the threaded connection.

IMPROVEMENTS FOR BOLT PRE-TENSION

The **Bolt Pre-Tension** feature has been added to the Solid Mechanics interface also in 2D axial symmetry. In this case, a bolt is by necessity located at the axis of revolution.

In the **Bolt Selection** subnode, you can now prescribe a relaxation. This can, for example, be a function of time and loading history.

COUPLING BEAMS AND SOLIDS

The **Solid-Beam Connection** multiphysics coupling is now available in 3D. Three fundamentally different types of connection can be modeled:

- A point on a beam is connected to a boundary (or part of a boundary) on the solid. The connected region is rigidly coupled to the point on the beam.
- An edge with a beam representation is connected to a boundary on a solid. All nodes on the solid that are within a certain transverse distance from the beam are connected.
- A transition from a beam to a solid is modeled. In this case, the solid is assumed to be a 3D representation of the beam cross section, and beam theory assumptions are used when formulating the connection. Even the warping of the cross section is taken into account.

In 2D, one more type of connection has been added: **Solid and beam shared boundaries**.

IMPROVED DEFAULT PLOTS

The default plots in the Structural Mechanics physics interfaces have been generally improved.

- The color table for default von Mises stress plots have been changed to **RainbowLight**.
- Mode shape plots for eigenfrequency and linear buckling studies now use the **AuroraBorealis** color table.
- Mode shape plots have the legend switched off in order to emphasize that the amplitude of a mode does not have a physical meaning.
- Section force plots in the Beam and Truss interfaces use a symmetric **Wave** color scale. This makes it possible to immediately distinguish between tension and compression, for example.
- In contact analysis, a plot of the contact pressure is added, either as a line plot (2D) or a contour plot (3D).

- The default plot for **Stress Linearization** now has a legend for the graphs.
- The default **Undeformed geometry** plot, produced by the Shell interface, has new colors.

IMPROVED PLOTS FOR PRINCIPAL VALUES

The plot type **Principal Stress** can now be used for any kind of tensor principal values. In earlier versions, only a single predefined stress or strain field could be selected. Now it is also possible to manually enter orientation vectors and corresponding principal values.

A new set of principal strains have been added to the results in the Solid Mechanics interface: Principal logarithmic strains. This is the logarithmic (or “true”) strain, with orientations given in a space-fixed coordinate system. This quantity is thus suited for plotting on the deformed geometry in a geometrically nonlinear analysis.

Also, a new **Principal Stress Line** plot type has been added. Previously, the principal value plots were available only for volumes and surfaces. The new plot type is particularly useful in the Shell and Plate interfaces.

THERMAL EXPANSION MULTIPHYSICS COUPLING PROVIDES HEAT SOURCE

In the **Thermal Expansion** multiphysics coupling, you can choose to let energy dissipation variables, such as work done by plastic deformation, directly act as source terms for the Heat Transfer in Solids interface. To do that, select the **Mechanical Losses** check box in the **Heat Sources** section.

EIGENFREQUENCY ANALYSIS AND GEOMETRIC NONLINEARITY

When performing a prestressed eigenfrequency analysis, the spatial and material frames are now different. This has the following implications:

- It is now possible to perform an eigenfrequency analysis, linearized around a certain state in a contact analysis.
- All kinds of follower loads will be taken into account. As an example, a force that is given in a deformation-dependent coordinate system will be treated as a follower load.
- A side effect is that results may be different in old models where anything that changes with deformation was referenced. The default boundary system is one such case; an extrusion coupling operator defined as having a spatial frame is another.

MODAL PARTICIPATION FACTORS

The participation factor computation has been completely redesigned. The new implementation has two significant improvements:

- It is now possible to compute participation factors for structures in which a combination of physics interfaces is used.
- Rotational participation factors are now available.

To compute modal participation factors, you add a **Participation Factors** node under **Definitions>Variable Utilities**.

C AND HAT CROSS SECTIONS IN THE BEAM INTERFACE

Two more built-in cross section types have been added to the Beam interface: C and Hat. You can access them under **Common Sections** in the **Cross Section Data** node.

WARPING COMPUTED IN THE BEAM CROSS SECTION INTERFACE

You can now compute the actual warping of a beam cross section, given the axial twist of the beam.

ADDITIONS TO SAFETY FACTOR COMPUTATIONS

The **Safety** feature has been augmented in two respects:

- In the Membrane interface, the Modified Tsai-Hill, Norris, Azzi-Tsai-Hill, Hoffman, Tsai-Wu orthotropic, and Tsai-Wu criteria have been added.
- Failure criteria for concrete (Bresler-Pister, Willam-Warnke, and Ottosen) have been added for the Solid Mechanics, Shell, Plate, and Beam Cross Section interfaces.

NORMAL AND SHEAR STRESS

Normal stress and shear stress components of traction and their magnitudes have been added as variables in the Solid Mechanics interface.

DETAILED CONTROL OVER CONSTRAINTS

All constraints in the Structural Mechanics interfaces have been augmented by a possibility to exclude the constraints on objects on a lower dimension. As an example, a **Prescribed Displacement** on a boundary can now be disabled on a set of edges or points. This is useful when you need to fine-tune your constraints; for example, when there are duplicates or conflicts between constraints.

WEAK FORM OF AUTOMATIC CONSTRAINTS IN THE SHELL INTERFACE

Since the rotational degrees of freedom have different meanings across a fold line, they are separate degrees of freedom, which are joined by a constraint. This constraints is, by default, implemented as a pointwise constraint. Now you can choose to use a weak constraint instead, by a control in the **Fold-Line Limit Angle** section in the settings for the Shell interface.

In order to maintain the property that the shell normal has unit length, a constraint is applied on the shell normal displacement degrees of freedom in each node. This constraints is, by default, implemented as a pointwise constraint. Now you can choose to use a weak constraint instead, by a control in the **Advanced** section in the settings for the Shell interface.

IMPROVEMENTS IN THE SHELL INTERFACE

In the Shell interface, a number of improvements to the boundary conditions have been made:

- The formulation of the **Antisymmetry** constraint has been revised.
- The formulations of **Prescribed Velocity** and **Prescribed Acceleration** in a geometrically nonlinear analysis have been revised.
- **Prescribed Displacement/Rotation** for points and boundaries have been changed so that the input of rotation is analogous to the same boundary condition on edges.
- All edge boundary conditions, like **Edge Load**, now include a line scale factor that incorporates the effect of curvature as an offset. The effect is that the contribution accounts for the fact that an edge of the true midsurface may have a length that is different from the length of the edge in the geometry.
- Many boundary conditions can take orientations from local shell coordinate systems and have a new section: **Face defining the local orientation**.
- The default plot in the Shell interface now uses a Gauss-point-evaluated stress. The default expression, which makes use of a new general operator, is `gpeval(4,shell.mises)`.

Backward Compatibility with Version 5.2

The friction slip velocity is no longer defined as a dependent variable. When running a model using the COMSOL API, you will need to remove the reference to this variable in the solver settings. The friction slip velocity is usually defined as `<comp>_<solid>_vslip_<pairname>`, where `<comp>` is the tag of the component,

`<solid>` is the tag of the Solid Mechanics physics interface where the **Friction** node is defined, and `<pairname>` is the name of the contact pair.

Backward Compatibility with Version 4.3b

- If an old model using the Beam interface is opened, the Euler-Bernoulli formulation is used. The Beam Formulation selection is still shown but cannot be changed from Euler-Bernoulli. If Advanced Physics Options is enabled, then the new section **Backward Compatibility** is shown. If you clear the **Use pre 4.4 formulation** check box, then the new formulation is used. Doing this enables the use of Timoshenko beams, but you must manually handle solver settings like segregation and scaling. This legacy option does not support beams mixed with solids or shells in cases where the same names for degrees of freedom were used in both physics interfaces.
- For models using mechanical contact, plot expressions are not updated when you open an old model where the pair names have been edited manually. Also, opening an old model where a pair name has been manually edited might affect the solver configuration. It is recommended that you regenerate a new default solver configuration.

Backward Compatibility with Version 4.3a

- When using version 5.2a, in the Solid Mechanics interface, the property `ControlsSpatialFrame` from version 4.3a no longer exists, and COMSOL Multiphysics now always assumes that the displacements control the spatial frame. Remove all lines setting the value of `ControlsSpatialFrame`. See Support Knowledge Base article 1156.
- When opening old models that include contact, the penalty factor control is set to **User defined** and the contact normal penalty factor is set to the expression used in the model.

Subsurface Flow Module

New Functionality in Version 5.3a

NEW FREE AND POROUS MEDIA INTERFACE

The Free and Porous Media interface has been updated. The new interface is of the modern style, based on a multiphysics coupling. With the new approach, all functionality in the constituent physics interfaces is available. Old MPH-files will run with the old interface, but a warning will be displayed.

NEW KOZENY-CARMAN PERMEABILITY MODEL IN THE DARCY'S LAW AND RICHARDS EQUATIONS INTERFACES

The Darcy's Law and Richards Equations interfaces can now define the permeability from the so-called Kozemy-Carman equation. The Kozeny-Carman permeability model allows you to estimate the permeability of granular soils and packed beds from the main particle diameter and porosity.

NEW CUBIC LAW MODEL OF TRANSMISSIVITY IN THE FRACTURE FLOW INTERFACE

The Fracture Flow interface can now define the fracture's permeability from the so-called cubic law equation. The cubic law determines the transmissivity of the fracture from the aperture and roughness factor.

NEW THIN BARRIER NODE IN THE TWO-PHASE DARCY'S LAW INTERFACE

The Two-Phase Darcy's Law interface can now define permeable walls on interior boundaries. These internal boundaries are used to represent thin, low-permeability structures. The Thin Barrier feature allows you to avoid meshing thin structures like geotextiles or perforated plates. The permeability of the internal wall can be either isotropic or anisotropic.

IMPROVEMENTS TO THE THIN BARRIER NODE IN THE DARCY'S LAW AND RICHARDS EQUATIONS INTERFACES

The Darcy's Law and Richards Equations interfaces can now define isotropic or anisotropic permeable walls on interior boundaries.

New Models in Version 5.3a

PERMEABILITY ESTIMATION MODEL

This tutorial model demonstrates the capability of COMSOL Multiphysics to estimate the permeability of a porous material by creating a detailed fluid flow model in a small unit cell. The computed permeability can be used in homogenized models using Darcy's law or the Brinkman equations.

GEOTHERMAL DOUBLET MODEL

This tutorial model demonstrates the application of COMSOL Multiphysics to model a geothermal doublet. A geothermal doublet is used to produce hot groundwater, extract heat and reinject the cooled-down water into the subsurface. The extracted heat can be used for domestic heating or converted into electric energy.

SOLUTE INJECTION MODEL

This tutorial model demonstrates the application of COMSOL Multiphysics to a benchmark case of subsurface flow and solute transport. The model tracks a contaminant that enters an aquifer at a point, such as an injection well or toxic spill, and spreads through the aquifer with time. The model results are compared to the analytic solution.

SOLUTE TRANSPORT MODEL

This tutorial model demonstrates the application of COMSOL Multiphysics to a benchmark case of subsurface flow and transient solute transport. The simulation tracks the solute transport over an area of 16 square km during 1000 days in groundwater flow, accounting for longitudinal and transversal dispersivities. The results are compared to analytical solutions in a benchmark.

Wave Optics Module

New Functionality in Version 5.3a

- The Electromagnetic Waves, Frequency Domain and the Electromagnetic Waves, Beam Envelopes physics interfaces now have physics-controlled meshing enabled by default. Physics-controlled meshing now handles more general frequency-dependent material properties and the Drude-Lorentz, Debye, and Sellmeier dispersion models. Physics-controlled meshing can now be used in mode analysis simulations.
- For the Electromagnetic Waves, Beam Envelopes interface, the default setting is to created a swept mesh in 3D and a mapped mesh in 2D, with an additional option of creating a tetrahedral or triangular mesh in 3D and 2D, respectively.
- For frequency- and wavelength-dependent study steps used together with any of the Electromagnetic Waves, Frequency Domain or Electromagnetic Waves, Beam Envelopes physics interfaces, a default wavelength of 1 μm is now used.
- A new Gaussian beam background field implementation is now available, where the Gaussian beam is defined using a summation of plane waves propagating with wave vectors pointing in a distribution around the main propagation direction. Since a plane wave is a solution to the Helmholtz equation, the plane-wave expansion is a solution to the Helmholtz equation, whereas the existing paraxial implementation is only an approximation to the Helmholtz equation. Thus, the plane-wave expansion can also be used for tightly focused Gaussian beams.
- New postprocessing variables for reflectance, transmittance, and absorptance have been introduce. The variables are named `ewfd.Rport_`*<x>* and `ewfd.Tport_`*<x>*, where *<x>* is the port name. In addition, when periodic and diffraction order ports are used, there are also new variables based on the mode number and mode polarization. For instance, the variable `ewfd.Rorder_p1_op` represents the reflectance for a 2D mode with order 1 and out-of-plane polarization. In addition, there are summation variables for the total reflectance, total transmittance, total reflectance and transmittance, and absorptance.
- For frequency or wavelength sweeps, parametric sweeps, and auxiliary sweeps, new default plots of the reflectances and transmittances are generated. When the study

does not perform any sweeps, instead of a default plot, a global evaluation into a table is created for the reflectance and transmittance variables.

- When a boundary mode analysis is performed, new variables for the effective indices of the modes are created. The name for those variables follows the pattern `<phys>.neff_<x>`, where `<phys>` is the physics interface tag and `<x>` is the port name.

New Model in Version 5.3a

A new *Slot Waveguid*e model has been added to the Wave Optics Module Application Library.

Backward Compatibility with Version 5.2 and Earlier

PERFECTLY MATCHED LAYERS PHYSICS FEATURE NO LONGER AVAILABLE

The **Perfectly Matched Layer** physics feature that for version 5.2 and earlier versions was marked deprecated is no longer available. Thus, models using the **Perfectly Matched Layer** physics feature, must first be rebuilt in a version earlier than 5.2a to use the **Perfectly Matched Layer** feature from under the **Definitions** node, before being loaded in 5.2a.

NEW DEFAULT VALUE FOR GAUSSIAN BEAM BACKGROUND FIELD SPOT RADIUS

The Gaussian beam background field is based on the paraxial approximate solution to Helmholtz' equation. Since this approximation is less accurate for small spot radii, the default value for the spot radius has been changed from one wavelength to ten wavelengths.

PORT PHASE PARAMETER RENAMED AND RELOCATED

The parameter Port phase, which was previously located in the **Port Properties** section, is now called Mode phase and is located in the **Port Mode Settings** section. The API, however, has not changed.

BOUNDARY MODE STUDY SEQUENCES NO LONGER AVAILABLE IN THE MODEL WIZARD

Previously, when the Laser Heating multiphysics was selected, the Boundary Mode, Frequency-Stationary and Boundary Mode, Frequency-Transient study sequences were available from the Model Wizard. Now, these two study sequences are removed. Instead, a Boundary Mode Analysis step can be added manually to the Frequency-Stationary and Frequency-Transient studies.

Material Library

New and Updated Material Data in Version 5.3a

In version 5.3a, 150 new materials and 1300 new material properties have been added. In addition, these improvements have been made:

- Added s-N fatigue data for cast iron 80-55-06 and A356.0, expansion data for Mo-35V and Mo-29Ti alloys, stress-strain curves for Au-5Pd-2Ni braze foil, and thermal conductivity for Inconel 702 and 347 stainless.
- The tensile data for 416 stainless was incorrect. The data was for the tempering temperature, not the test temperature. The data has been replaced with the correct values.
- The tensile data for 17-4 PH stainless (only condition H1100) has been replaced with a new data source. The previous source is believed to be incorrect.
- Data has been added for 718 Plus, Invar, 17-4 PH, 301, 321, 347, and 9 Ni steel.
- The specific heat data for 718 Plus was off by a factor of 4.148 and has been corrected.
- The thermal conductivity of NbC was too high by a factor of 100 and has been corrected.
- A new reference for the expansion of Al 5754 is used; the values has changed by approximately 3%.
- Additional data for the expansion of H10 has been added; the values has changed by approximately 5%.
- New references for the specific heat of Rene-41 and 17-4 are used. The new data fits with other similar alloys better than the data from the previous references. The data below about 300 degrees C for 17-4 PH and below 500 degrees C for Rene-41 is significantly different.
- An additional reference for the expansion data for Astroloy has been added, and the data was refitted. The data below 300 degrees C differs by approximately 4%.
- The flexural modulus, flexural strength, and apparent viscosity have been added to the database as new properties.

LiveLink™ for MATLAB®

New Functionality in Version 5.3a

LiveLink™ *for* MATLAB® now supports MATLAB® R2017a and R2017b.

The following sections list new and improved functionality in LiveLink™ *for* MATLAB®.

CONNECTION TO THE COMSOL SERVER

An installation of LiveLink™ *for* MATLAB® with the COMSOL Client is now available to connect the MATLAB® Client to a COMSOL Server™. The supported modes of operation include the use of MATLAB® functions in apps, and establishing a connection from MATLAB® to COMSOL Server™ to access the COMSOL API.

IMPROVEMENTS TO ADDING FUNCTIONS DEFINED BY M-FILES IN MATLAB® TO MODELS

Functions defined under Global Definitions in a model, including those functions that are defined by M-files in MATLAB®, are now available for use in all text fields (and table cells) that previously supported the use of global parameters.

NEW AND IMPROVED WRAPPER FUNCTIONS

mphreduction Function

The new `mphrreduction` function makes it possible to extract reduced-order state-space matrices from a time-dependent model. The extracted state-space matrices can be used for simulation, control design, and other types of model analysis.

mphplot Function

Using the `mphplot(model)` function, you can now generate a MATLAB® figure window where you can easily switch between geometry, mesh, and result plots in the model. With buttons on the toolbar, you can switch on/off lights and switch between views. The camera position is also remembered when switching between different plots in the figure window.

mphmesh Function

The function `mphmesh` has improved support for views and supports hiding of geometric entities.

mphthumbnail Function

The wrapper function `mphthumbnail` can now directly use the graphics displayed in a MATLAB® figure window for creating a model thumbnail that can be saved when the model file is saved. It is now also possible to clear a thumbnail if desired.

Backward Compatibility with Version 5.2

The functions `mpheval`, `mphglobal`, and `mphinterp`, which extract numerical data, now support complex values by default; that is, the property `'complexout'` is `'on'` by default. If you want to have the previous behavior, set `'complexout'` to `'off'`.

Backward Compatibility with Version 5.0

- The `mphint2` command: For data sets other than Solution, Particle, Cut Line, Cut Plane, Time Integral, Time Average, Surface, and Line, the integration order now corresponds to the element refinement. The default value is set to 4 instead of 1. This change can affect the solution in cases where the problem is not resolved well enough.
- The `mphapplicationlibraries` function replaces the `mphmodellibrary` function for showing the Application Libraries window.

LiveLink™ for Excel®

New Functionality in Version 5.3a

In order to make it easier to work with plots and the data that defines the plot, it is now possible to export data from 1D plots to Excel® with LiveLink™ *for* Excel®. The new button, placed in the **Numerical Results** section of the **COMSOL** ribbon tab in Excel®, extracts the numerical data from the selected plot and creates a corresponding chart in Excel®. The chart generated in Excel® is an XY Scatter chart.

The COMSOL API for Use with Java®

COMSOL 5.3a API Changes

PARTITION OBJECTS PROPERTIES

The `keep` property has been deprecated in version 5.3a. Instead, use the new properties `keepinput` and `keeptool`. If you set `keep` to on, both `keepinput` and `keeptool`. If you get the value of `keep`, it is `on` if `keepinput` and `keeptool` are `on`.

COMSOL 5.3 API Changes

NEW COMPONENT SYNTAX

The geometry component is now included in the code for features that appear in a component. For example,

```
model.geom("geom1")
```

in earlier versions is now instead

```
model.component("comp1").geom("geom1")
```

to indicate which component the geometry belongs to. To keep the old syntax in generated Model Java® files, clear the **Use component syntax** check box under **Code generation** on the **Methods** page in the **Preferences** window. All old scripts and Java® source code files will continue to work using the old syntax.

ECAD IMPORT MODULE CHANGES

- The default value for the `sellayer` property of the `Import` function is now `on`. Previously, the default was `off`.
- The default value for the `sellayershow` property of the `Import` function is now `all`. Previously, the default was `dom`.

COMSOL 5.2a API Changes

AC/DC MODULE CHANGES

New Coil Feature

In the Magnetic Fields, Magnetic and Electric Fields, and Rotating Machinery, Magnetic interfaces, a new unified **Coil** feature has replaced the **Multi-Turn Coil** feature. The functionality of the old **Multi-Turn Coil** feature can be replicated by using the new **Coil** feature with the Conductor model set to Homogenized Multi-Turn, but all of the other parameters are unchanged. The only change required is to the create statements such as:

```
model.physics("mf").feature().create("mtcd1",
"MultiTurnCoilDomain", 3);
```

which should be modified to:

```
model.physics("mf").feature().create("mtcd1", "Coil", 3);
model.physics("mf").feature("mtcd1").set("ConductorModel",
"Multi");
```

User-Defined Coil Geometry Subfeature Selection

In 3D models, when using a user-defined geometry for a **Coil** feature, the **User Defined Coil Geometry** subfeature has been transformed from a boundary feature to a domain feature. New **Input** and **Output** subnodes are used to select the coil input or output boundaries instead. Any code setting the selection of this feature should be modified to set the selection on the **Input** subfeature instead.

Relative Permeability for Magnetic Shielding in Magnetic Fields, No Current

The relative permeability parameter `mur` in the **Magnetic Shielding** feature in the Magnetic Fields, No Currents interface is now a scalar quantity. Setting a tensor quantity will not cause exception, but only the first component of the tensor will be used as the value.

ACOUSTICS MODULE CHANGES

The default settings have changed for the Incident Pressure Field in the Pressure Acoustics interfaces. Add the following line to obtain the old behavior of this feature when `ipf1` is a subfeature to `pwr1`:

```
  model.physics("acpr").feature("pwr1").feature("ipf1").
  set("c", "acpr.c_c");
```

COMSOL 5.1 API Changes

ACOUSTICS MODULE CHANGES

The ratio of specific heats is now a material parameter. If the default value was used in a model, add the following line to the Java® file to obtain the old behavior:

```
model.physics("cpf").feature("cpf1").set("gamma_mat", "userdef");
```

ELECTROCHEMISTRY MODULES CHANGES

There are changes for the Tertiary Current Distribution, Nernst-Planck interface. The Java® API syntax for creating and accessing vectors and tensors has changed, as well as the syntax for setting physics properties. See the following examples:

- Specifying the diffusion tensor.

  Old syntax:

  ```
  model.physics("tcdee").feature("ice1").set("D_0", new
  String[]{"D1", "0", "0", "0", "D1", "0", "0", "0", "D1"});
  model.physics("tcdee").feature("ice1").set("D_1", new
  String[]{"D2", "0", "0", "0", "D2", "0", "0", "0", "D2"});
  model.physics("tcdee").feature("ice1").set("D_2", new
  String[]{"D3", "0", "0", "0", "D3", "0", "0", "0", "D3"});
  ```

  New syntax (includes the species name in the first argument of the set method):

  ```
  model.physics("tcdee").feature("ice1").set("D_c1", new
  String[]{"D1", "0", "0", "0", "D1", "0", "0", "0", "D1"});
  model.physics("tcdee").feature("ice1").set("D_c2", new
  String[]{"D2", "0", "0", "0", "D2", "0", "0", "0", "D2"});
  model.physics("tcdee").feature("ice1").set("D_c3", new
  String[]{"D3", "0", "0", "0", "D3", "0", "0", "0", "D3"});
  ```

- Accessing tensor components (in the definitions of other variables, for example).

  Old syntax:

  ```
  Dxx_c1, Dxy_c1, Dxz_c1...
  ```

  New syntax (component index after the species name):

  ```
  D_c1xx, D_c1xy, D_c1xz...
  ```

- Specifying the reaction rate vector.

  Old syntax:

  ```
  model.physics("tcdee").feature("reac1").set("R_0", new
  String[]{"R1"});
  model.physics("tcdee").feature("reac1").set("R_1", new
  String[]{"R2"});
  ```

```
model.physics("tcdee").feature("reac1").set("R_2", new
String[]{"R3"});
```

New syntax (includes the species name in the first argument of the set method):

```
model.physics("tcdee").feature("reac1").set("R_c1", new
String[]{"R1"});
model.physics("tcdee").feature("reac1").set("R_c2", new
String[]{"R2"});
model.physics("tcdee").feature("reac1").set("R_c3", new
String[]{"R3"});
```

- Accessing vector components.

  Old syntax:

  ```
  tfluxx_c1, tfluxy_c1, tfluxz_c1
  ```

  New syntax (component index after the species name):

  ```
  tflux_c1x, tflux_c1y, tflux_c1z
  ```

- Specifying initial values.

  Old syntax:

  ```
  model.physics("tcdee").feature("init1").setIndex("c2", "c20", 0);
  model.physics("tcdee").feature("init1").setIndex("c3", "c30", 0);
  model.physics("tcdee").feature("init1").setIndex("V", "V0", 0);
  ```

  New syntax:

  ```
  model.physics("tcdee").feature("init1").setIndex("initc", "c20",
  1);
  model.physics("tcdee").feature("init1").setIndex("initc", "c30",
  2);
  model.physics("tcdee").feature("init1").setIndex("initphil",
  "V0");
  ```

- Selecting properties.

  Old syntax:

  ```
  model.physics("tcdee").prop("Convection").set("Convection", 1,
  "0");
  model.physics("tcdee").prop("ConvectiveTerm").setIndex("Convectiv
  eTerm", "noncons", 0);
  ```

  New syntax (the property name corresponds to the section in the user interface):

  ```
  model.physics("tcdee").prop("TransportMechanism").set("Convection
  ", 1, "0");
  model.physics("tcdee").prop("AdvancedSettings").setIndex("Convect
  iveTerm", "noncons", 0);
  ```

For information about API backward compatibility for the Surface Reactions interface, see the Chemical Reaction Engineering Module release notes.

ECAD IMPORT MODULE CHANGES

- The default value for the `grouping` property of the `Import` function is now set to `layer`. Previously, the default was `all`.
- For ODB++® and ODB++(X) files (the `type` property is `odb`), new rules apply for the initialization of the `importlayer` property of the `Import` function. Now, only layers of the types *Metal* and *Dielectric* are initialized with the string `on` in the `importlayer` string array. Previously this also included layers of type *Drill*.

LIVELINK™ FOR SOLID EDGE® CHANGES

The default value for the `keepfree` property of the `LiveLinkSolidEdge` function is now set to `on`. Previously, the default was set to `off`.

LIVELINK™ FOR SOLIDWORKS® CHANGES

The default value for the `keepfree` property of the `LiveLinkSOLIDWORKS` function is now set to `on`. Previously, the default was set to `off`.

COMSOL 5.0 API Changes

DOCUMENTATION

The *COMSOL Multiphysics Programming Reference Manual* replaces the *COMSOL API for Use with Java® Reference Manual.*

DEPRECATED METHODS

The following methods were deprecated in COMSOL 5.0:

TABLE 1-3: DEPRECATED METHODS AND THEIR REPLACEMENTS

DEPRECATED METHOD	NEW METHOD
`com.comsol.model.OptFeature.field()`	`OptFeature.comp()`
`com.comsol.model.OptFeature.field(String)`	`OptFeature.comp(String)`
`com.comsol.model.Group.identifier()`	`Group.paramName()`
`com.comsol.model.Material.identifier()`	`ModelEntity.tag()`
`com.comsol.model.MaterialModel.identifier()`	`ModelEntity.tag()`
`com.comsol.model.ModelNode.identifier()`	`ModelEntity.tag()`

TABLE 1-3: DEPRECATED METHODS AND THEIR REPLACEMENTS

DEPRECATED METHOD	NEW METHOD
`com.comsol.model.physics.Physics.identifier()`	Use `tag()` instead.
`com.comsol.model.physics.MultiphysicsCoupling.identifier()`	`ModelEntity.tag()`
`com.comsol.model.Group.identifier(String)`	`Group.paramName(String)`
`com.comsol.model.Material.identifier(String)`	`ModelEntity.tag(String)`
`com.comsol.model.MaterialModel.identifier(String)`	`ModelEntity.tag(String)`
`com.comsol.model.ModelNode.identifier(String)`	`ModelEntity.tag(String)`
`com.comsol.model.physics.Physics.identifier(String)`	Use `tag(string)` instead.
`com.comsol.model.physics.MultiphysicsCoupling.identifier(String)`	`ModelEntity.tag(String)`
`com.comsol.model.ModelEntity.name()`	`ModelEntity.label()`
`com.comsol.model.ModelEntity.name(String)`	`ModelEntity.label(String)`
`com.comsol.model.ParameterEntity.set(String, int, double)`	Use the 0-based version (`setIndex`) instead.
`com.comsol.model.ParameterEntity.set(String, int, double)`	Use the 0-based version (`setIndex`) instead.
`com.comsol.model.ParameterEntity.set(String, int, double[])`	Use the 0-based version (`setIndex`) instead.
`com.comsol.model.ParameterEntity.set(String, int, int)`	Use the 0-based version (`setIndex`) instead.
`com.comsol.model.ParameterEntity.set(String, int, int[])`	Use the 0-based version (`setIndex`) instead.
`com.comsol.model.ParameterEntity.set(String, int, int, double)`	Use the 0-based version (`setIndex`) instead.
`com.comsol.model.ParameterEntity.set(String, int, int, int)`	Use the 0-based version (`setIndex`) instead.
`com.comsol.model.ParameterEntity.set(String, int, int, String)`	Use the 0-based version (`setIndex`) instead.

TABLE 1-3: DEPRECATED METHODS AND THEIR REPLACEMENTS

DEPRECATED METHOD	NEW METHOD
`com.comsol.model.ParameterEntity.set(String, int, String)`	Use the 0-based version (`setIndex`) instead.
`com.comsol.model.ParameterEntity.set(String, int, String[]`	Use the 0-based version (`setIndex`) instead.

COMSOL 4.4 API Changes

PHYSICS INTERFACES

- The default **Initial Value** features in the following interfaces have been extended:
  - Transport of Concentrated Species
  - Reacting Flow in Porous Media (rfcs)
  - Reacting Flow
  - Rotating Machinery, Reacting Flow

  Previously, only the initial mass fractions could be specified. Now, the initial values can be specified in terms of mass fractions, mole fractions, molar concentrations, number densities, or densities. When opening an old Model MPH file, **Mixture specification** is set to **Mass fractions**, and the mass fractions specified are entered in the respective text fields. In old model files for Java®, commands specifying the initial value of a mass fraction return an error in version 5.0 and later. To specify the initial mass fraction of a species `w2`, which is the second in the list of species (seen in the **Dependent Variables** section when clicking the interface), change the COMSOL API Java® command

  ```
  model.physics("chcs").feature("init1").set("w2", 1, "0.1");
  ```

  to

  ```
  model.physics("chcs").feature("init1").set("w0", 2, "0.1");
  ```

- The Pressure Acoustics, Frequency Domain interface and the Boundary Mode Acoustics interface have a new default feature. If the default feature has been edited in a Java® file, the following line should be added to the Java® file to obtain the old behavior for Pressure Acoustics, Frequency Domain:

  ```
  model.physics("acpr").feature().create("pam1",
  "PressureAcousticsModel").selection().all();
  ```

  or

```
model.physics("acbm").feature().create("pam1",
"PressureAcousticsModel").selection().all();
```

for Boundary Mode Acoustics.

This line should be added directly after the line where the physics interface is created.

- For models using mechanical contact, the variable field name in the solver settings for the old contact variables is different than in earlier versions. For example:
  - In version 5.2 and later: `mod1_solid_contact_old_p1`, `mod1_solid_cm_old_p1`.
  - In version earlier than 5.0: `mod1_solid_contact_p1_old`, `mod1_solid_cm_p1_old`.
- The parameter form in the property `EquationForm` can no longer be set to `ModeAnalysis` in 3D and 2D axisymmetric calculations for the Pressure Acoustics, Frequency Domain and Pressure Acoustics, Transient interfaces. In these cases, the parameter `modeFreq` has been removed.

COMSOL 4.3b API Changes

PHYSICS INTERFACES

- The *Crosswind diffusion* formulation has been updated for all physics interfaces supporting Crosswind diffusion. Models solved with the new formulation can give different results than models solved in versions earlier than 4.3b. Java® files can be modified to retain old crosswind formulations. Please contact COMSOL Support for details.
- The default settings have changed for the Background Pressure Field in the Pressure Acoustics interfaces. Add the following line to obtain the old behavior of this feature:

  ```
  model.physics("acpr").feature("bpf1").set("c", 1, "acpr.c_c");
  ```
- In the Acoustics Module, the **Far-Field Calculation** feature is no longer available in 1D and 1D axisymmetric models.
- The shape function property `border` has been deprecated and replaced by `order`.

COMSOL 4.3a API Changes

GENERAL API CHANGES

- Class files compiled with COMSOL Multiphysics version 4.3 or earlier need to be recompiled if they contain code that changes the contents of `model.selection()`.
- The `solutionintitle` property governs whether to show solution information in plot titles. But since solution information has never been shown for 1D plots, regardless of the content of the property, the property has been removed from 1D plot groups.

FLUID FLOW INTERFACES API

- Fluid flow features in the CFD Module and Microfluidics Module that previously created a scalar Lagrange multiplier field with default name `model.un_lm` now create a vectorial Lagrange multiplier field with default name `model.u_lm`. The default component names in 3D are `model.u_lm`, `model.v_lm`, and `model.w_lm`, respectively. Java® files must be updated accordingly.
- Weak constraints for the Interior Wall feature are no longer available. Any references to its weak constraint parameter (`weakConstraints`) or Lagrange multipliers must be removed.
- The **Automatic** setting for the **Pseudo time stepping** property now sets the variable *<phtag>*`.locCFL` to the built-in variable `CFLCMP`, which in turn triggers a PID regulator via the automatic solver suggestions. Here, *<phtag>* is the physics interface tag. Java® files where pseudo time stepping is active and the local CFL number is set to **Automatic** must be modified by adding the command

  ```
  model.physics(<tag>).prop("PseudoTimeProperty").set("CFLNumbExpr"
  , 1, "Manual");
  ```

  before calling a solver.

COMSOL 4.3 API Changes

- In the Acoustics Module, the far-field variables `pfar` and `Lp_far` have new names with full scope. They are now referred to as `acpr.ffc1.pfar` and `acpr.ffc1.Lp_far`, respectively. Any model files for Java® that use the old variable names in expressions (such as expressions used for plotting or evaluation that include such old variable names) require a manual update.
- The following methods

  ```
  model.physics(<tag>).feature(<ftag>).params();
  ```

```
model.physics(<tag>).prop(propname).params();
```

are deprecated and replaced by the methods

```
model.physics(<tag>).feature(<ftag>).param();
model.physics(<tag>).prop(propname).param();
```

COMSOL 4.2a API Changes

- Class files compiled with COMSOL Multiphysics version 4.1 or earlier need to be recompiled.
- The **Far Field** feature in the Electromagnetic Waves interface has changed from being a boundary feature with a boundary selection only, to a domain feature with a domain selection. It also has a subfeature — a boundary selection.
- The dependent variable associated with gauge fixing in the Magnetic Fields, Magnetic and Electric Fields, Electromagnetic Waves, and Transient Electromagnetic Waves interfaces is now unique to each interface. It is no longer available in the model scope, for example, `mod1.psi`. Instead, the gauge fixing field is only available in the interface scope, for example, as `mod1.mf.psi`.
- In the scattered field formulation in the Electromagnetic Waves interface, the scattered field is no longer available in the model scope (for example, `mod1.relEx`). Instead, the scattered field is only available in the interface scope as, for example, `mod1.emw.relEx`.
- In the Solid Mechanics interfaces (`SolidMechanics`, `PiezoelectricDevices`, `AcousticStructure`, `Poroelasticity`, `ThermallyInducedStresses`, `JouleHeatingThermalExpansion`, `TransientAcousticStructure`, and `FluidStructureInteraction`), tensors in local coordinate systems (`el`, `eel`, `Sl`, `si`, and `ei`) have new names. The coordinates of the local system (for example, `x1`, `x2`, and `x3`) are no longer used. Instead 1, 2, and 3 are used together with double indexing for all components. As an example, `elX2` is replaced by `el22` and `elx2x3` is replaced by `el23`. The tensors `si` and `ei` are now called `Sil` and `eil`.
- In the Darcy's Law interface and the Richards' Equation interface in the Subsurface Flow Module, fluid compressibility is now a material parameter and no longer has a default value. If the default value was used, you now have to set the value. The following example sets the permeability to the old default value:

```
model.physics("dl").feature("smm1").set(chif_mat,userdef);
model.physics("dl").feature("smm1").set(kappa,4e-10);
```

- In the Poroelasticity interface in the Subsurface Flow Module, the fluid compressibility and the Biot-Willis coefficient are now material parameters and no longer have default values. If the default values were used, you now have to set the value. The following example sets the permeability to the old default value:

```
model.physics("dl").feature("smm1").set(chif_mat,userdef);
model.physics("dl").feature("smm1").set(kappa,4e-10);
```

- The Level Set and Phase Field interfaces now include the **Initial Interface** feature by default. If you have a model that was created in an earlier version of COMSOL Multiphysics, it will fail to create a feature with the same tag name.

COMSOL 4.1 API Changes

The following changes were made to the COMSOL API between versions 4.0a and 4.1:

- In version 4.0a, the property `nonlin` in the stationary solver could have the values `auto`, `on`, `off`, and `linearized`. The default in most cases was `auto`, but in some cases, specifically in a frequency-domain study, the default was `linearized`. In version 4.1, the possible values are `auto`, `on`, `off`, and `linper`, with `auto` as default, except in the Frequency Domain, Linearized study step where the default is `linper`. The value `linearized` can still be set, but this is treated as `auto`. If the problem is linear and the linearization point is zero, `auto` should give the same solution as `linearized`. For nonlinear problems where the linearization point is zero, the solution given by `auto` might be slightly different, and probably more accurate, while changing to `off` should give exactly the same solution as `linearized`. When there is a nonzero linearization point, it is relevant to use the `linper` option. This option works like `linearized`, except that source terms that contribute to the linearized problem must be enclosed by the `linper` operator.
- In the Darcy's Law interface, Brinkman interface, Richards' Equation interface, and Poroelasticity interface (in the CFD Module and Subsurface Flow Module), the permeability and porosity are now material parameters and no longer have default values. If the default values were used in a model file for Java®, you now have to set them. The following example sets the permeability to the old default value:

```
model.physics("dl").feature("dlm1").set(kappa_mat,userdef);
model.physics("dl").feature("dlm1").set(kappa,3e-11);
```

- The pressure variable in solid mechanics, typically `solid.pw`, now only gets allocated degrees of freedom for incompressible materials. If you have referred to it, for example, during segregation in the solver, the code will have to be changed to remove the field.

- The solution to eigenvalue and eigenfrequency problems may now appear in a different order.
- The Batteries & Fuel Cells Module's boundary feature, `BoundaryReactionCoefficient`, is tagged by `rc` instead of `brc` by default. This means that files that explicitly refer to the old default tag name must be modified to refer to the new tag name.

COMSOL 4.0a Java API Changes

The following changes were made to the COMSOL API between versions 4.0 and 4.0a:

- The units of the load face parameter (`Fph`) of the **Phase** feature in `AcousticStructure`, `TransientAcousticStructure`, `ThermallyInducedStresses`, `SolidMechanics`, `JouleHeatingThermalExpansion`, `Poroelasticity`, and `FluidStructureInteraction` have changed from degrees to radians.
- The `physics.field()` operator for `Beam` and `Truss` are not backward compatible with 4.0. Scalar fields are now vector fields.
- The variables for strain and stress in the Shell interface have been renamed. Variable names that began with a lowercase `s` now begin with an uppercase `S`. For variables names that ended in lowercase `l` (L), the letter was moved to the third position. For example, `emXXl` is now `emlXX`.
- Force loads on solids and trusses have been moved from the material to the spatial frame. This means that the index notation of some variables has changed from uppercase `XYZ` to lowercase `xyz`.
- The acoustics interfaces (`PressureAcoustics`, `TransientPressureAcoustics`, `AcousticStructure`, and `TransientAcousticStructure`) have been moved from the material to the spatial frame. This means that the index notation of vector variables has changed from uppercase `XYZ` to lowercase `xyz`.

This concludes the release notes for COMSOL Multiphysics version 5.3a.

Index

A AC/DC Module
- backward compatibility 35
- new application 35
- new functionality 34

Acoustics Module
- backward compatibility 40
- new functionality in 38

Application Builder. new functionality in 10

B backward compatibility, general considerations 23

Battery & Fuel Cells Module
- backward compatibility 44
- new functionality in 42

C CAD Import Module
- new functionality in 45

CFD Module
- backward compatibility 48
- new functionality in 47
- new models in 48

Chemical Reaction Engineering Module
- backward compatibility 56
- new functionality in 55
- new models in 55

COMSOL API changes 133

COMSOL Multiphysics, new functionality in 13

COMSOL Server, new functionality in 13

Corrosion Module
- backward compatibility 61
- new and updated models in 60
- new functionality 60

D Design Module
- new functionality in 45

E ECAD Import Module
- backward compatibility 62

Electrochemistry Module
- backward compatibility 63
- new functionality in 63

Electrodeposition Module
- backward compatibility 65
- new and updated models in 64
- new functionality in 64

F Fatigue Module
- backward compatibility 66

G general new functionality 10

Geomechanics Module
- new functionality in 67

geometry and mesh, new functionality for 16

H hardware-accelerated image export 21

Heat Transfer Module
- backward compatibility 74–76
- new applications in 72
- new functionality in 68

L LiveLink for Excel
- new functionality in 132

LiveLink for MATLAB
- backward compatibility 131
- new functionality in 130

LiveLink™ products for CAD
- new functionality in 45

M Material Library
- new and updated material data in 129
- new material data in 129

MEMS Module
- backward compatibility 86
- new functionality in 86

Microfluidics Module

backward compatibility 88
Mixer Module
new functionality in 89
Molecular Flow Module
backward compatibility 90
Multibody Dynamics Module
new applications in 91
new functionality in 91

N Nonlinear Structural Materials Module
new applications in 94
new functionality in 94

O operators, functions, and definitions, new and updated 17

P Particle Tracing Module
backward compatibility 97
new applications in 97
new functionality in 96
Pipe Flow Module
backward compatibility 101
Plasma Module
backward compatibility 102–103

R Ray Optics Module
backward compatibility 106
new applications in 106
new functionality in 105
results and visualization, new functionality in 21
RF Module
new and updated apps and models in 112
new functionality in 111
Rotordynamics Module
new applications in 115
new functionality in 114

S Semiconductor Module
new applications in 117
new functionality in 116
Structural Mechanics Module
backward compatibility 122
new functionality in 118
studies and solvers, new functionality in 18
Subsurface Flow Module
new functionality in 124
new models in 125

W Wave Optics Module
backward compatibility 127
new functionality in 126
new model in 127