LiveLink™ *for* Revit®

User's Guide

5.5

COMSOL

LiveLink™ *for* Revit® User's Guide

Version: COMSOL 5.5

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: CM024501

Contents

Chapter 1: Introduction

Chapter 2: The LiveLink™ Interface

Chapter 3: Geometry Tools and Features

Chapter 4: Programming and Command Reference

1

Introduction

Welcome to the LiveLink™ *for* Revit® User's Guide! This guide details the functionality of this optional package that extends the COMSOL Multiphysics® modeling environment with additional tools and features to use geometry from the Revit® building information management software for simulation, to import and export geometry using the most common 3D CAD file formats, and to repair, defeature, and modify geometry.

This introductory chapter contains an overview of the capabilities of the module, including a summary of the included geometry features, an overview of this guide, and a description of where to find documentation and model examples.

About the Product

Overview of the Included Geometry Tools and Features

LiveLink™ *for* Revit® enables modeling using 3D designs synchronized from the Revit® building design software. The included user interface builds on an associative transfer of the geometry from the CAD program to the COMSOL model.

If you rather use file import to get your designs into COMSOL Multiphysics, the product also supports import of the most common 3D CAD file formats: *ACIS*®, *AutoCAD*®, *IGES*, *Inventor*®, *NX*®, *Parasolid*®, *PTC*® *Creo*® *Parametric*™, *PTC*® *Pro/ENGINEER*®, *SOLIDWORKS*®, and *STEP*. In addition, support for *CATIA*® *V5* is available as a separate add-on. To exchange data with CAD packages, you can export your geometry to the *ACIS*®, *IGES*, *Parasolid*®, and *STEP* file formats.

Finally, the product provides a dedicated geometric kernel, the *CAD kernel*, and a wide range of tools for you to prepare an imported 3D design for meshing and analysis. You can interactively search for and remove geometric features, for example, fillets, holes, slivers, small faces, and short edges. You can also modify objects by detaching a portion to form an additional computational domain, or by creating a fluid domain for computation, in case the CAD design only includes the solid parts.

GEOMETRY FEATURE	**ICON**	**DESCRIPTION**
Cap Faces		Generate faces from edges to fill gaps and create solid objects, or to partition solids
Convert to COMSOL		Convert to the COMSOL kernel representation
Delete Faces		Delete and replace faces
Delete Fillets		Find and delete fillets
Delete Holes		Find and delete holes
Delete Short Edges		Find and delete short edges
Delete Sliver Faces		Find and delete sliver faces
Delete Small Faces		Find and delete small faces

GEOMETRY FEATURE	ICON	DESCRIPTION
Delete Spikes		Find and delete spikes from faces
Detach Faces		Detach faces and form a new object from them
Export		Export geometry objects to 3D CAD file formats
Import		Import geometry objects from 3D CAD file formats
Knit to Solid		Knit surface objects to form solid or surface object
LiveLink for Revit		Synchronize geometry between Revit and COMSOL
Repair		Repair and removal of small details

Overview of the User's Guide

This documentation covers and the add-on for file import of CATIA® V5 files. Instructions on how to use the geometry modeling tools in COMSOL Multiphysics® in general are included with the *COMSOL Multiphysics Reference Manual*. To help you get started with modeling this module is also accompanied by the quick-start guide *Introduction to* .

Where Do I Access the Documentation and Application Libraries?

A number of internet resources have more information about COMSOL, including licensing and technical information. The electronic documentation, topic-based (or context-based) help, and the application libraries are all accessed through the COMSOL Desktop.

> **!** If you are reading the documentation as a PDF file on your computer, the blue links do not work to open an application or content referenced in a different guide. However, if you are using the Help system in COMSOL Multiphysics, these links work to other modules (as long as you have a license), application examples, and documentation sets.

THE DOCUMENTATION AND ONLINE HELP

The *COMSOL Multiphysics Reference Manual* describes all core physics interfaces and functionality included with the COMSOL Multiphysics license. This book also has instructions about how to use COMSOL Multiphysics and how to access the electronic Documentation and Help content.

Opening Topic-Based Help

The Help window is useful as it is connected to many of the features on the GUI. To learn more about a node in the Model Builder, or a window on the Desktop, click to highlight a node or window, then press F1 to open the Help window, which then displays information about that feature (or click a node in the Model Builder followed by the **Help** button ( ). This is called *topic-based* (or *context) help*.

Win	To open the **Help** window: • In the **Model Builder**, **Application Builder**, or **Physics Builder** click a node or window and then press F1. • On any toolbar (for example, **Home**, **Definitions**, or **Geometry**), hover the mouse over a button (for example, **Add Physics** or **Build All**) and then press F1. • From the **File** menu, click **Help** ( ). • In the upper-right corner of the COMSOL Desktop, click the **Help**( ) button.

Mac Linux	To open the **Help** window: • In the **Model Builder** or **Physics Builder** click a node or window and then press F1. • On the main toolbar, click the **Help** ( ) button. • From the main menu, select **Help>Help**.

Opening the Documentation Window

Win	To open the **Documentation** window: • Press Ctrl+F1. • From the **File** menu select **Help>Documentation** ( ).

Mac Linux	To open the **Documentation** window: • Press Ctrl+F1. • On the main toolbar, click the **Documentation** ( ) button. • From the main menu, select **Help>Documentation**.

THE APPLICATION LIBRARIES WINDOW

Each application includes documentation with the theoretical background and step-by-step instructions to create a model application. The applications are available in COMSOL as MPH-files that you can open for further investigation. You can use the step-by-step instructions and the actual applications as a template for your own modeling and applications. In most models, SI units are used to describe the relevant properties, parameters, and dimensions in most examples, but other unit systems are available.

Once the Application Libraries window is opened, you can search by name or browse under a module folder name. Click to view a summary of the application and its properties, including options to open it or a PDF document.

The Application Libraries Window in the *COMSOL Multiphysics Reference Manual*.

Opening the Application Libraries Window

To open the **Application Libraries** window ( ):

Win

- From the **Home** toolbar, **Windows** menu, click ( ) **Applications Libraries**.
- From the **File** menu select **Application Libraries**.

To include the latest versions of model examples, from the **File>Help** menu, select ( ) **Update COMSOL Application Library**.

Mac

Linux

Select **Application Libraries** from the main **File>** or **Windows>** menus.

To include the latest versions of model examples, from the **Help** menu select ( ) **Update COMSOL Application Library**.

CONTACTING COMSOL BY EMAIL

For general product information, contact COMSOL at info@comsol.com.

To receive technical support from COMSOL for the COMSOL products, please contact your local COMSOL representative or send your questions to support@comsol.com. An automatic notification and case number is sent to you by email.

COMSOL WEBSITES

COMSOL website	www.comsol.com
Contact COMSOL	www.comsol.com/contact
COMSOL Access	www.comsol.com/access
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

2

The LiveLink™ Interface

When running the COMSOL Multiphysics® software and the Revit® building information management software side-by-side you can transfer the geometry of room volumes, architectural elements, and conceptual masses from Revit® to COMSOL Multiphysics using the LiveLink™ interface. In the other direction, from COMSOL Multiphysics to Revit®, the interface enables you to update the dimensions of conceptual masses in the Revit® project.

You may also analyze designs using simulation apps that connect to Revit® by utilizing the LiveLink interface. With the provided tools you can easily connect to COMSOL Server™ from within Revit® to browse and run simulation apps, including those that use geometry synchronized with the CAD software.

This section includes the following topics:

- Synchronizing the Geometry
- Connecting to COMSOL Server™ and Running Applications

Synchronizing the Geometry

To initiate the geometry synchronization between Revit® and COMSOL Multiphysics® use the **LiveLink for Revit** feature node.

Before synchronization consider to review and change the settings for the LiveLink node, and to configure the synchronization of architectural elements present in the Revit® project as described in this section.

This section includes the following topics:

- The LiveLink Node, where you initiate a synchronization
- The Synchronization Settings Window is where you can manage the configurations for synchronizing geometry from projects inside Revit®
- The Settings Window for Configurations in Revit® is where you can select the elements to be synchronized in a configuration

The LiveLink Node

The **LiveLink for Revit** feature, available from the **LiveLink** menu on the **Home** toolbar, synchronizes the geometry between Revit® and COMSOL Multiphysics®.

For geometry synchronization to take place both COMSOL Multiphysics and Revit® need to be running, and the CAD file needs to be open in the CAD software.

> A list of compatible versions of Revit® can be found at: www.comsol.com/system-requirements/module.

ASSOCIATIVITY AND GEOMETRY OPERATIONS

During synchronization the LiveLink interface generates and transfers the geometry objects for the volumes of selected rooms in the Revit® project, and retrieves and transfers the geometry of architectural elements, including masses. Mass objects are rebuilt before transfer based on the parameter values set in the COMSOL Multiphysics model. To ensure that associativity is preserved use selections for the architectural elements to apply model settings, for example material and physics settings.

In the geometry sequence of the model the LiveLink node signifies a geometry synchronized from the CAD software, and in many aspects it is just like any other geometry operation. It can be combined with other operations that may appear both before and after the LiveLink node in the sequence.

Synchronized geometry objects are represented using the Parasolid® geometry kernel inside COMSOL Multiphysics. Thus, you can apply all the tools and features for defeaturing and geometry modification as included with this LiveLink™ product.

THE SYNCHRONIZE SECTION

To specify which project to synchronize use the **Synchronize with** list. Select **Active document** to synchronize the open and active project in Revit®. For the initial synchronization after adding a LiveLink™ node, **Active document** is the only available setting. For subsequent synchronizations the **Specified document** option becomes selected instead. Using this setting the project specified after **Document** will be synchronized provided that it is open in Revit®. To synchronize a new project switch to the **Active document** setting. The name of the project is automatically determined during synchronization with the **Active document** setting.

When running parametric optimization studies or parametric sweeps, the project needs to remain open in Revit® until the solver completes.

The LiveLink interface also determines the selected **Configuration** in the synchronized Revit® project when **Synchronize with** is set to **Active document**. To make sure that the project will be synchronized in the same state during subsequent synchronizations, the **Specified document** alternative can be used. With this option the interface automatically activates the last synchronized configuration. To be able to synchronize the project in a different configuration, first make the desired configuration active in Revit®, then from the **Synchronize with** list select **Active document**.

For information on how to set up synchronization configurations for a Revit® project see the section titled The Synchronization Settings Window.

You can also select to **Synchronize material data** together with the geometry. With this option the interface imports the material properties defined in the synchronized part or assembly components, and the software creates corresponding **Material** nodes in the component. The input selection in the generated **Material** nodes is set to the material selections that are also created during synchronization; see Selections, for more information.

To initiate a synchronization click the **Synchronize** button. This sends parameter value pairs to Revit®, then retrieves the regenerated geometry objects. Synchronization can also be triggered by clicking a build button when there are changed settings in the LiveLink node, or there are changed parameter values. Synchronization is automatic when running an optimization study or a parametric sweep.

PARAMETERS

Parameters that take part in the synchronization are specified in the **Parameters in CAD Package** section. Based on the transferred parameter and value pairs in the **Controllable parameters** table, the CAD model is automatically rebuilt in Revit® and transferred back to COMSOL Multiphysics®. This way you can control mass parameters in the Revit mass model.

Together with the updated geometry, updated values of the parameters listed in the **Read-only parameters** table are also transferred from the CAD software. Read-only parameters are parameters in Revit® that are defined by a formula and therefore not possible to control without breaking the CAD design. However these parameters can be used to set up the simulation.

Clicking **Synchronize** also retrieves parameters that not yet appear in the tables under the **Parameter in CAD Package** section but have been selected to be linked to COMSOL® from the Revit® project. As part of this process a corresponding global parameter is automatically generated in the model.

In both the **Controllable parameters** and **Read-only parameters** tables, the **CAD name** column holds the names of mass parameters in the Revit® project, whereas the **COMSOL name** and **COMSOL value** columns contain the name and value, respectively, of corresponding global parameters in the model, defined under **Global Definitions>Parameters** in the model tree. Global parameters can be controlled by the parametric solver to perform parametric sweeps. During synchronization of controllable parameters COMSOL evaluates the corresponding global parameters and sends the resulting value to Revit®. Click the symbol in the **Sync** column to turn on or off the synchronization of a parameter.

Unless a unit is specified the updated parameters are assumed to have units as defined in the Revit® project.

You can type in parameters in the tables, or use the **COMSOL Parameter Selection** window in Revit® to link parameters from the Revit® project, for details see the section The COMSOL Parameter Selection Window. Linked parameters can then be retrieved to the table, and global parameters are automatically generated for them.

Synchronizing Parameters

To retrieve the linked parameters from the Revit® project to the LiveLink node, and to generate corresponding global parameters in the model do one of the following:

- In the **Settings** window for **LiveLink for Revit** click the **Synchronize** button:

  New parameters, which have been selected in the Revit® project, but are not listed under **Parameters in CAD Package**, are transferred to the **Controllable parameters** table or the **Read-only parameters** table. For each new mass parameter a global parameter is generated in the model. The global parameters are assigned the values of the corresponding mass parameters. Following this, the geometry is regenerated in Revit®, based on the parameters in the **Controllable parameters** table, and transferred to COMSOL.

- In the **Settings** window for **LiveLink for Revit** click the **Update Parameters from CAD** button ( )

  New parameters, which have been selected in the Revit® project, but are not listed under **Parameters in CAD Package** are transferred to the **Controllable parameters** table or the **Read-only parameters** table. For each new mass parameter a global parameter is generated in the model. The global parameters are assigned the values of the corresponding mass parameters. In addition, the values of global parameters, which are already linked to mass parameters in the table, are updated to the current values of the mass parameters.

IMPORT OPTIONS

In the **Length unit** list, select **From COMSOL** to scale the transferred objects to the length unit of the geometry in the current model. Select **From CAD document** to change the geometry's length unit to the unit in the CAD software.

Objects to Import

Select the types of objects to transfer from Revit® using the **Solids**, **Surfaces** check boxes.

Import Options

The **Absolute import tolerance** is a length measured in the geometry's unit after synchronization. The import operation merges geometric entities with a distance smaller than this tolerance.

If the **Check imported objects for errors** check box is selected, a warning appears if the transferred objects contain errors.

If the **Repair imported objects** check box is selected, the software tries to repair defects and remove details smaller than the **Absolute import tolerance** when transferring objects from Revit®.

If the **Remove redundant edges and vertices** check box is selected, edges and vertices that are considered redundant, such as the edges of an imprint on a face, are removed during synchronization.

SELECTIONS

The LiveLink™ interface synchronizes selections for selected elements from the Revit® project. The selections get their names from the element name, type and category. Synchronized selections appear in the **Selections from CAD Package** table. Click on an entry in the table to see the included objects highlighted in the **Graphics** window. Selections are available in all applicable selection lists, for example in geometry and mesh operations, or material and physics settings, but do not appear as separate selection nodes in the COMSOL model tree. See the section The Settings Window for Configurations on how to set up the synchronization to include selections for the various elements.

The Synchronization Settings Window

To configure the synchronization of a project between Revit® and COMSOL click the **Synchronization Settings** button on the **COMSOL Multiphysics** tab in Revit®. The **Synchronization Settings** window that opens contains the tools for creating and managing multiple collections of synchronization settings, referred to as *configurations*. A configuration contains information about which view or rooms that should be synchronized, how the room volume should be determined, and how to synchronize room bounding and other elements in the room.

The **Configurations** table holds all configurations in the project. To edit the name of a configuration double click it in the **Name** column. The radio button in the **Active** column controls which configuration is used when the project is synchronized between Revit® and COMSOL. To set up a synchronization configuration click the **Edit** button to open a dialog box that contains the settings for the configuration, see The Settings Window for Configurations.

To add a new configuration to the table click **New**.

Click **New from View** to create a configuration based on an existing *View* from the Revit® *Project Browser*. See New from View.

To create a copy of a configuration, select it from the table, then click **Duplicate**.

Select a configuration from the table, then click **Preview** to open a separate window with a graphical representation of the configuration. This is useful to check the elements included in the synchronization.

Click **Delete** to discard a configuration from the table. **Clear All** deletes all configurations from the table.

Select the **Save settings in project** check box to store the synchronization configurations in the Revit® project. Next time the project is saved the configurations will be saved in the file. Clear the **Save settings in project** check box to remove stored configurations from the Revit® project file next time you save the project.

Click **OK** to confirm changes and close the window, or click **Cancel** to discard changes and close the window. Note that synchronization is not possible if the **Synchronization Settings** window is open in Revit®.

NEW FROM VIEW

To set up a synchronization configuration based on elements visible in an existing view in the Revit® project click **New from View** in the **Synchronization Settings** window.

The **Select View** dialog box opens where you can select the view to use for the configuration. Supported views include *Floor Plans* and *3D Views*. Select a view and click **OK** to confirm and close the dialog box, or click **Cancel** to cancel the operation.

The configuration created from a view includes the room volumes (for floor plan views) and elements visible in the selected view. For example a floor plan view of a level in the building includes all the rooms on the level together with all visible elements.

The Settings Window for Configurations

In the **Synchronization Settings** window, click the **Edit** button for a configuration. The window that opens contains the settings for the configuration.

Select the **Synchronize view** check box to synchronize elements contained in the 3D view selected from the **Select view** list. The **Create selection** check box is selected by default to generate selections for the synchronized elements. This mode of synchronization is useful for projects without any room definitions and when the elements are not contained in rooms. The type of elements that can be synchronized include for example wall layers, structural framing and foundation elements, mechanical and electrical equipment.

Clear the **Synchronize view** check box to enable the synchronization of room volumes, elements contained in rooms, and conceptual masses. To synchronize both room volumes and the type of elements that are only available when synchronizing 3D views you can create two configurations, one for the room volumes and another for the 3D view. Then in the geometry sequence of your COMSOL® model you can add two LiveLink™ nodes for separate synchronization of each configuration.

ROOM SETTINGS

The **Rooms** table displays a list of rooms that are defined in the Revit® project. To turn on the synchronization of a room select its check box from the list.

> Room synchronization may fail or produce unexpected results if the room is not defined by room bounding elements. In the Revit® project check that walls, floors, ceilings and roofs are set to be *Room Bounding*, and that *Room Separators* are defined where necessary.

Room Extension

Select **Extend to walls** to generate a room volume that extends to the room bounding elements, such as walls, interior walls, floors, windows, and ceilings or roofs. The room volume is transferred to the COMSOL model as a solid object.

Using the extend to wall option for rooms results in the geometry objects for adjacent rooms to be disconnected unless solids for room bounding elements are also synchronized. Select the **Solids for room bounding elements** check box under the **Synchronize room bounding elements** section.

With the **Extend to wall centerline** option the surface objects for the room bounding elements are generated at the centerline of walls, and at the top plane of floors, ceilings and roofs. Note that with this option a solid object for the room volume is not generated during synchronization, only surface objects are transferred. In the COMSOL model, after synchronization create a solid using either the **Convert to Solid** or **Knit to Solid** operations with the surfaces as input.

Element Settings

The **Elements** table contains a list of elements contained in the rooms that are selected in the **Rooms** table. By default all available elements are shown in the table, to filter the display of elements by category use the **Show category** list box.

A selected check box for an element in the **Synchronize** column means that the geometry for the element will be transferred to COMSOL during synchronization. In the COMSOL model this can result in one or multiple solid or surface objects for the element, depending on how the element is represented in the Revit® project.

The setting in the **Detail level** column determines how the geometry for the element is generated before synchronization. The following options are available:

- **Bounding box** (default): An enclosing box is generated for the element.
- **Original**: The geometry of the element is transferred without alterations.

Elements may be represented only by surfaces in the Revit project and they may also have small details that may result in problems during mesh generation. Transferring elements with **Detail level** set to **Original** may therefore require the geometry to be modified and/or simplified in COMSOL after synchronization.

Select the check box in the **Create selection** column for the synchronization to create a selection in the COMSOL model for the element. After synchronization the selections appear in the **LiveLink for Revit** settings window, in the **Selections from CAD Package** table. The selections are available in all applicable selection lists but do not appear as separate selection nodes in the COMSOL model tree.

Synchronization of Room Bounding Elements

With the check boxes in the **Synchronize Room Bounding Elements** section you can control the type of geometry that is synchronized for room bounding elements such as walls, floors, ceilings, roofs, windows. The following options are available:

- **Solids for room bounding elements**: Select this check box to synchronize solid objects for room bounding walls, floors, ceilings and roofs. Solid objects are generated for all walls, including interior walls, which are adjacent to selected rooms.
- **Surfaces for room bounding elements**: Select this check box to synchronize surface objects for room bounding walls, floors, ceilings and roofs.
- **Surfaces for room interior walls**: Select this check box to synchronize the midsurfaces of interior walls to the room. This setting is enabled only when the **Extend to wall centerline** option is selected for the room extension.
- **Surfaces for openings**: Select this check box to synchronize surface objects for windows, skylights, curtain walls, and doors. This option also generates surfaces for openings that do not have any elements defined.
- **Create selections**: Select this check box to create selections in the COMSOL model for the geometry objects representing the walls, floors, windows, ceilings and roofs.

After synchronization the selections appear in the **LiveLink for Revit** settings window, in the **Selections from CAD Package** table. The selections are available in all applicable selection lists but do not appear as separate selection nodes in the COMSOL model tree.

Conceptual Masses

Conceptual masses in a project are synchronized when the check box **Synchronize masses** is selected. Synchronization of masses can be enabled independently of the synchronization of rooms. To include the mass floors in the mass synchronization select the **Include mass floors** check box. Selections for masses in the COMSOL model are automatically generated when the **Create selection** check box is selected.

The COMSOL Parameter Selection Window

In Revit®, open the **COMSOL Parameter Selection** window by clicking the **Parameter Selection** button ( ) located on the **COMSOL Multiphysics** tab. The window lists all mass parameters in the active project. To link a parameter to COMSOL select the corresponding check box in the **Add to COMSOL** column. All types of dimensions can be selected, but only dimensions that are not defined by a formula are possible to control from a COMSOL model. Linking dimensions that are defined by a formula enables using their values in COMSOL model definitions.

Select the **Save settings in project** check box to save the list of linked parameters in the file.

Connecting to COMSOL Server™ and Running Applications

Overview

A COMSOL® runnable application is a COMSOL Multiphysics® MPH-file that, in addition to the model part, includes a custom user interface that you can run as a separate application or in a web client. For simulations with geometry that comes from a Revit® model, applications can also use the LiveLink™ interface for Revit®.

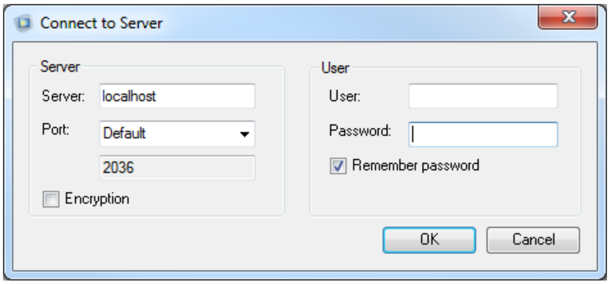
To create such applications, use the Application Builder, which is included in the Windows® version of COMSOL Multiphysics. For more information on how to build applications, refer to the book *Introduction to Application Builder*.

With a COMSOL Server™ license together with a license for LiveLink™ *for* Revit®, a COMSOL application that uses the LiveLink interface can be run by connecting to a COMSOL Server with an easy-to-install COMSOL Client, available for download from www.comsol.com. The software components installed with the COMSOL Client allow you to connect to a COMSOL Server right from the user interface of Revit®, and to browse and launch apps using the COMSOL Server interface. COMSOL Server or COMSOL Client does not include the Application Builder, Physics Builder, and Model Builder tools that come with the COMSOL Desktop® environment, and that are required for building applications.

For more information on the installation and administration of the COMSOL Server, refer to the book *COMSOL Server Manual*.

Connecting to a COMSOL Server™

To log in to a COMSOL Server™ interface, on the **COMSOL Multiphysics** tab in Revit® click the **COMSOL Server** ( ) button. After you enter a valid username and password



the COMSOL Server interface is displayed embedded in the Revit® user interface. Here you can view the apps available in the Application Library.

To launch an app click the **Launch** button below its icon. The app is launched in a separate COMSOL Client window. The first time you start the COMSOL Client you will need to log in to the COMSOL Server.

As an alternative to connecting to the COMSOL Server interface from Revit®, you can also start the COMSOL Client from the Windows® Desktop or **Start** menu, and log in to the COMSOL Server to launch apps.

Note: Applications that use LiveLink™ *for* Revit® require a COMSOL Client installation as they are not supported to be run from a web browser.

Running Applications with COMSOL Multiphysics®

If you have a COMSOL Multiphysics® installation you can still launch apps from the Revit® user interface by clicking the **Run Application** ( ) button on the **COMSOL Multiphysics** tab in Revit®. In the **Open** dialog box browse to the application, then click **Open**. This will bring up the app interface in a separate window. If the application utilizes the LiveLink™ interface make sure that the CAD document is open in Revit® before using the app.

Note that the **Run Application** button is disabled if you have a COMSOL Client installation of LiveLink™ *for* Revit®. In this case run the app with the COMSOL

Client, for example by first logging in to a COMSOL Server interface as described in the section Connecting to a COMSOL Server™.

3

Geometry Tools and Features

This chapter describes the tools and features available for importing and modifying geometry with LiveLink™ *for* Revit®.

Geometry Representation

Working with the CAD Kernel

The component of the COMSOL Multiphysics® software that is used to represent, build, and manage the interactions between geometric objects is the geometric kernel or geometric modeler. There are two kernels used by the software, the *COMSOL kernel*, and the *CAD kernel* (the Parasolid® kernel) that is included with the CAD Import Module, the Design Module, and LiveLink™ products interfacing CAD packages.

With a license for LiveLink™ *for* Revit® the software defaults to the CAD kernel for representing the geometry. You need to use the CAD kernel to apply the geometry features included with this module, for example the defeaturing and repair tools, as well as to import 3D geometries using various 3D CAD file formats.

The 3D operations and primitives listed in Table 3-1 do not support the CAD kernel — they always use the COMSOL kernel. However, an automatic conversion is performed for these objects before they are used as input to geometry features that require the CAD kernel, see Converting Objects to CAD Kernel Representation.

TABLE 3-1: 3D GEOMETRY FEATURES THAT DO NOT SUPPORT THE PARASOLID GEOMETRY KERNEL

FEATURE NAME	FEATURE NAME
Bezier Polygon	Point
Eccentric Cone	Polygon
Extrude	Pyramid
Helix	Revolve
Hexahedron	Sweep
Interpolation Curve	Tetrahedron
Parametric Curve	Torus
Parametric Surface	Work Plane

CHANGING THE GEOMETRIC KERNEL

To switch between geometric kernels, you can click the **Geometry** node, then in its Settings window, from the **Geometry representation** list choose either the **CAD kernel** or **COMSOL kernel**.

When you change the **Geometry representation** setting, all nodes that support the CAD kernel are marked as edited with an asterisk (*) in the upper-right corner of the node's icon. To rebuild the geometry using the new kernel, click the **Build All** button ( ). To avoid re-solving an already solved model, you can click the **Update Solution** button ( ) on the **Study** toolbar to map the solutions from the geometry represented by the CAD kernel to the new geometry represented by the COMSOL kernel.

If you solve a model using the CAD kernel, it is not possible to view and postprocess the solution if you open it in a COMSOL Multiphysics session where a license for the CAD Import Module, Design Module, or one of the LiveLink for CAD products is not available, unless, before saving the model, you change the geometry representation to COMSOL kernel and update the solution. This is possible to do only for 3D geometry sequences that do not contain geometry features that require the CAD kernel.

When you create a new model, its default geometry representation is controlled by the preference setting **Geometry>Geometry representation>In new models**.

When you open an existing model, you normally use the geometry representation used in the model. To always get the possibility to convert the geometry to the COMSOL kernel, change the preference setting **Geometry>Geometry representation>When opening an existing model** to **Convert to COMSOL kernel**.

Converting Objects to COMSOL Kernel Representation

To convert CAD objects (geometric objects represented by the CAD kernel) to objects represented by the COMSOL kernel, from the **Geometry** toolbar, **Conversions** menu, select **Convert to COMSOL** ().

The COMSOL geometry file format (.mphbin, or .mphtxt) can contain geometric objects saved in both the CAD kernel and COMSOL kernel representations. To import geometry from such a file to a geometry sequence that uses the COMSOL kernel, you need to convert geometry objects to the COMSOL representation before exporting to the file.

CONVERT TO COMSOL

Select the objects that you want to convert in the Graphics window. The selected objects are displayed in the **Input objects** list.

SELECTIONS OF RESULTING ENTITIES

If you want to make the resulting entities contribute to a cumulative selection, select a cumulative selection from the **Contribute to** list (the default, **None**, gives no contribution), or click the **New** button to create a new cumulative selection (see Cumulative Selections in the *COMSOL Multiphysics Reference Manual*).

Select the **Resulting objects selection** check box to create predefined selections (for all levels — objects, domains, boundaries, edges, and points — that are applicable) in subsequent nodes in the geometry sequence. To also make all or one of the types of resulting entities (domains, boundaries, edges, and points) that the resulting objects consist of available as selections in all applicable selection lists (in physics and materials settings, for example), choose an option from the **Show in physics** list: **All levels**, **Domain selection**, **Boundary selection**, **Edge selection**, or **Point selection**. The default is **Domain selection**, which is suitable for use with materials and physics defined in domains. For use with a boundary condition, for example, choose **Boundary selection**. These selections do not appear as separate selection nodes in the model tree. Select **Off** to not make any selection available outside of the geometry sequence.

Converting Objects to CAD Kernel Representation

If the current geometry representation for the geometry sequence is **CAD kernel**, an automatic conversion of COMSOL objects to CAD objects takes place before using the objects in Boolean operations and before using the objects in the **Convert to Solid, Convert to Surface, Convert to Curve**, and **Convert to Point** operations. This ensures that the CAD kernel is used in the above mentioned operations. This conversion is also performed when COMSOL objects are used as input to features that require the CAD kernel, for example the **Knit to Solid** feature

An automatic conversion to CAD objects is also performed before exporting geometry in the ACIS®, Parasolid®, STEP, and IGES file formats.

If the automatic conversion cannot be performed, the geometry operation is performed by the COMSOL kernel. For example, geometry objects created from a mesh cannot be converted to CAD kernel representation. Other examples of geometry objects that cannot be converted to CAD representation include objects that have an edge adjacent to three or more isolated faces, or objects that have a face bounded by an edge loop that intersects itself.

The automatic conversion to CAD kernel representation is not performed if one of the input objects to the Boolean or conversion operation is the result of a previous **Convert to COMSOL** operation.

Importing and Exporting CAD Files

Importing 3D CAD Files

To import geometry objects from a 3D CAD file, from the **Home** or the **Geometry** toolbar, click **Import** ( ). In the **Import** section of the Settings window, select **3D CAD file** from the **Geometry import** list. You can also skip this step as the type of the selected file is automatically recognized by the code. Click **Browse** to locate the file to import, or enter the path to the file. Before clicking the **Import** button consider to review and configure the import settings. If you have changed some settings after importing a file, the file is automatically re-imported when you click a build button.

The imported geometry objects are represented by the CAD kernel, see Working with the CAD Kernel, which is the geometric kernel used by the CAD Import Module, Design Module, and LiveLink™ products interfacing CAD packages.

Some 3D CAD formats use periodic parameterization for edges and faces. For example, a full-revolution cylindrical edge or face appears seamless in the CAD program. During import edges or faces that have a periodic parameterization are cut in two halves by inserting new vertices and edges. This is done because the mesh algorithms do not support periodic entities. You can ignore such inserted edges using an **Ignore Edges** feature from **Virtual Operations**.

SUPPORTED FORMATS

The CAD import supports the following 3D CAD formats:

TABLE 3-2: SUPPORTED 3D CAD FILE FORMATS

FILE FORMAT	NOTES	FILE EXTENSIONS	SUPPORTED VERSIONS
ACIS®	1,	`.sat`, `.sab`	`up to 2019 1.0`
AutoCAD®	1, 2	`.dwg`, `.dxf`	`2.5-2019`
CATIA® V5	2, 3	`.CATPart`, `.CATProduct`	`R8 to R2019`
PTC® Creo® Parametric™	1	`.prt`, `.asm`	`1.0-6.0`
IGES	1	`.igs`, `.iges`	`up to 5.3`
Inventor® assembly	1, 2	`.iam`	`11, 2008-2019`
Inventor® part	1, 2	`.ipt`	`6 to 11, 2008-2019`
NX®	1, 4	`.prt`	`up to 1847`

TABLE 3-2: SUPPORTED 3D CAD FILE FORMATS

FILE FORMAT	NOTES	FILE EXTENSIONS	SUPPORTED VERSIONS
Parasolid®	1	.x_t, .x_b	up to V32.0
PTC® Pro/ENGINEER®	1	.prt, .asm	16 to Wildfire 5
SOLIDWORKS®	1, 2, 5	.sldprt, .sldasm	98-2019
STEP	1	.step, .stp	AP203E1, AP214

Note 1: This format requires a license for one of the CAD Import Module, or Design Module, or LiveLink product for a CAD package.

Note 2: Available only on a supported Windows operating system.

Note 3: This format requires, in addition to the CAD Import Module, or Design Module, or a LiveLink product for a CAD package, a license for the File Import for CATIA V5 module.

Note 4: Support for the NX® file format is available only on a supported Windows or Linux operating system.

Note 5: Embedded parts in assemblies are not supported. To import such an assembly, first convert the embedded parts to external parts.

ASSOCIATIVITY

When possible the import maintains associativity for the imported geometry objects, so that when the CAD file is re-imported the settings applied to the geometric entities, for example physics or material settings, are retained. To maintain associativity the import relies on information in the CAD file that uniquely identifies the geometry objects and their entities, such as faces, edges, and points. This information is usually included in the CAD file if the geometry is saved in the format of the CAD software where it was created, but not when the geometry is exported to another CAD format. When re-importing a CAD file the import automatically tries to identify and match all geometry objects and their entities to the previous version. This may fail if the topology (structure) of the geometry has changed since the last import.

Note: To ensure that associativity is maintained when re-importing a CAD file work with CAD files saved in the originating CAD software's format, and avoid changes to the topology (structure) of the geometry. When an associative import is not possible use coordinate-based selections, such as the Ball, Box, and Cylinder selections in 3D (see Creating Selections From Geometric Primitives and Operations in the *COMSOL Multiphysics Reference Manual*).

LENGTH UNIT

In the **Length unit** list, select **From CAD document** to change the geometry's length unit to the unit in the file (if the file has a length unit). Select **From COMSOL** to keep the geometry's length unit and scale the objects in the file to the geometry's unit.

OBJECTS TO IMPORT

Select the types of objects to import using the **Solids**, **Surfaces**, and **Curves and points** check boxes.

If the **Surfaces** check box is selected, you can choose how COMSOL imports the surfaces using the list under **For surface objects**:

- Choose **Form solids** (the default) to knit together surface objects to form solids.
- Choose **Knit surfaces** to form surface objects by knitting.
- Choose **Do not knit** to not form any surface or solid objects from the imported surfaces.

For the **Form Solids** and **Knit surfaces** options select the **Fill holes** check box to generate new faces to replace missing geometry.

To import wireframe geometry you need to select the **Curves and points** check box. With this option, the **Unite curve objects** check box is selected by default to unite the imported curve objects, which speeds up the rendering of the geometry.

IMPORT OPTIONS

The **Absolute import tolerance** is a length measured in the geometry's unit after the import. When importing 3D CAD files, the program merges geometric entities with a distance smaller that this tolerance.

If you select the **Check imported objects for errors** check box, a warning appears if the imported objects contain errors.

If you select the **Repair imported objects** check box, the software tries to repair defects and remove details smaller than the **Absolute import tolerance**.

If you select the **Remove redundant edges and vertices** check box, the software tries to remove redundant edges and vertices.

SELECTIONS OF RESULTING ENTITIES

If you want to make the resulting entities contribute to a cumulative selection, select a cumulative selection from the **Contribute to** list (the default, **None**, gives no contribution), or click the **New** button to create a new cumulative selection (see Cumulative Selections in the *COMSOL Multiphysics Reference Manual*).

Select the **Resulting objects selection** check box to create predefined selections (for all levels — objects, domains, boundaries, edges, and points — that are applicable) in subsequent nodes in the geometry sequence. To also make all or one of the types of resulting entities (domains, boundaries, edges, and points) that the objects consist of available as selections in all applicable selection lists (in physics and materials settings, for example), choose an option from the **Show in physics** list: **All levels**, **Domain selection**, **Boundary selection**, **Edge selection**, or **Point selection**. The default is **Domain selection**, which is suitable for use with materials and physics defined in domains. For use with a boundary condition, for example, choose **Boundary selection**. These selections do not appear as separate selection nodes in the model tree. Select **Off** to not make any selection available outside of the geometry sequence.

Select the **Individual objects selections** check box to create predefined selections (for all levels — objects, domains, boundaries, edges, and points — that are applicable) in subsequent nodes in the geometry sequence for each individual object in the geometry file and for each relevant entity level. To also make all or one of the types of resulting entities (domains, boundaries, edges, and points) that the objects consist of available as selections in all applicable selection lists (in physics and materials settings, for example), choose an option from the **Show in physics** list: **All levels**, **Domain selection**, **Boundary selection**, **Edge selection**, or **Point selection**. The default is **Domain selection**, if available, which is suitable for use with materials and physics defined in domains. For use with a boundary condition, for example, choose **Boundary selection**. These selections do not appear as separate selection nodes in the model tree. Select **Off** to not make any selection available outside of the geometry sequence.

SELECTIONS GENERATED BASED ON INFORMATION IN THE CAD FILE

The following types of data from the CAD file are used to generate selection on the imported geometry:

- Material assignments can generate objects selections that are named according to the material names in the CAD file.
- Layer assignments of objects and entities, when supported by the CAD format, can generate object, boundary, edge, and point selections that are named according to the layer names in the CAD file.
- Color assignments to objects, faces, or edges can generate object, boundary, and edge selections, respectively.

After the import the generated selections are displayed in the Settings window for the Import node in sections named according to the entity level of the selections:

- **Object Selections**
- **Boundary Selections**
- **Edge Selections**
- **Point Selections**

Depending on which selections are generated, a subset of the above sections is displayed. The selections are listed in tables with the following columns:

- **Name**: Here you can edit the selection name that is generated by the import. For colors the generated names are of the type *Color 1*, *Color 2*, etc., for materials and layers the names from the CAD file are used.
- **Name in file**: This column contains the original name of the selection. To display this column select the **Show names from file** check box above the table.
- **Keep**: Select the check box in this column to make the selection available in selection lists for subsequent nodes in the geometry sequence.
- **Physics**: Select the check box in this column to make the selection available in all applicable selection lists (in physics and materials settings, for example).
- **Contribute to**: If you want to make the objects or entities in the selection contribute to a cumulative selection, select a cumulative selection from the **Contribute to** list (the default, **None**, gives no contribution), or click the **New Cumulative Selection** button under the table to create a new cumulative selection (see Cumulative Selections in the *COMSOL Multiphysics Reference Manual*).

Click a row in a table to highlight the corresponding selection on the geometry in the Graphics window. To help with identifying the color selections, these are highlighted with the colors defined in the imported CAD file. To always highlight on the geometry the color selections that you keep select from the **Graphics** toolbar **Colors > Show Selection Colors**.

The selections listed in the **Object Selections** section that are made available for the geometry sequence or physics setup are always available in all input selection lists, including all applicable entity selection lists. For example, the object selection of a solid object, generated for a material from the CAD file, automatically results in domain, boundary, edge, and point selections with the same name, so that you can use it to apply a boundary material, or a boundary condition. In contrast, a color assigned to a face of a solid object in the CAD file results in a boundary selection that is displayed in the **Boundary Selections** section, and it is available in all applicable boundary selection lists, but not, for example, in any edge selection lists.

Exporting Objects to 3D CAD Formats

With a license for LiveLink™ *for* Revit® you can export 3D geometry objects to the ACIS®, IGES, Parasolid®, and STEP formats. To do this:

- right-click the **Geometry** node and select **Export** ( ), or
- on the **Geometry** toolbar click **Export** ( ).

Then, in the **File type** list, select **Parasolid binary file**, **Parasolid text file**, **ACIS binary file**, **ACIS text file**, **IGES file**, or **STEP file**. Use the **Browse** button to choose the filename, then click **Save** to close the Export Geometry window.

Next, select **Export selected objects** to export only chosen geometry objects or select **Export entire finalized geometry** to export the resulting geometry of a Form Union or Form Assembly operation.

> Note that it is not possible to export to the formats mentioned here the result of virtual geometry operations that come after a Form Union or Form Assembly node in the geometry sequence.

When exporting to an ACIS file format choose the **ACIS file format version**. Available versions are **4.0**, **7.0**, **2016 1.0** (default).

For the Parasolid, IGES, and STEP file formats select a **Length Unit**. A unit conversion is carried out when the selected unit is different from the length unit of the geometry. A unit conversion is not done for the default **From geometry** option.

For the Parasolid file formats the option **Split in manifold objects** is selected by default to make sure that the exported geometry objects are manifold objects. A non-manifold object is, for example, a solid with an interior boundary that separates two domains. When exported using this option the solid is split along the interior boundary into two separate objects. When exporting to the ACIS, IGES, and STEP formats non-manifold objects are always split.

Finally, to export the geometry, click the **Export** button.

The Parasolid binary and text formats do not allow coordinate values larger than 500. Therefore you might have to change the export unit in the **Length unit** list box to be able to export the geometry.

COMSOL objects are automatically converted to CAD objects before saving the file.

For details on which objects can be converted to CAD objects see Converting Objects to CAD Kernel Representation.

Using the Defeaturing Tools

This section describes the defeaturing tools for removing details from imported 3D CAD geometry. With the defeaturing tools you can search for and delete both small details, such as short edges, small faces, sliver faces, and spikes, and larger details, for example, fillets, chamfers, and cylindrical holes.

To access the defeaturing tools, from the **Geometry** toolbar, **Defeaturing and Repair** menu, select **Delete Fillets**, **Delete Holes**, **Delete Short Edges**, **Delete Sliver Faces**, **Delete Small Faces**, **Delete Spikes**, **Delete Faces**, or **Detach Faces** from the submenu. You can also right-click the **Geometry** node and select the same options from the context menu.

When you are on the Tools window for a defeaturing operation, you can switch to another defeaturing tool by clicking one of the corresponding buttons at the top of the page. Upon completion of the defeaturing operation a corresponding feature node, which you can modify, appears in the geometry sequence.

Finding and Deleting Small Details

You can use any of the **Delete Fillets** ( ), **Delete Holes** ( ), **Delete Short Edges** ( ), **Delete Sliver Faces** ( ), **Delete Small Faces** ( ), and **Delete Spikes** ( ) tools to search for and delete details smaller than a given size. First activate the **Input objects** selection by clicking the **Active** button to toggle between and . Select the objects you want to examine in the Graphics window.

In the field **Maximum fillet radius**, **Maximum hole radius**, **Maximum edge length**, **Maximum face width**, **Maximum face size**, or **Maximum spike width**, enter the maximum size of the details you want to delete. When you click the **Find** button, a list of details that are smaller than the given size are shown in the list below. To delete the found details, either click the **Delete All** button, or select a subset of the found details in the list and click **Delete Selected**. Then, the selected details are deleted from their objects, and a node corresponding to this operation is added to the geometry branch of the model tree.

If you want to modify the performed deletion operation, you can select the added node in the geometry branch. Then, edit the node's form that appears in the **Settings** window. Click the **Build Selected** button ( ) to see the result of your edits.

Delete Faces

The **Delete Faces** ( ) page is used to delete faces and replace them either with a new face (if **Heal method** is **Fill**) or a by growing or shrinking the adjacent faces (if **Heal method** is **Patch**). Select the faces you want to delete in the Graphics window. They appear in the **Faces to delete** list. Select the **Heal as through hole** check box if you have selected faces that make up a hole that you want to delete. When you click the **Delete Selected** button, the selected faces are deleted, and a node corresponding to this operation is added to the geometry branch of the model tree.

If you want to modify the performed deletion operation, you can select the added node in the geometry branch. Then, edit the node's form that appears in the **Settings** window. Click the **Build Selected** button ( ) to see the result of your edits.

Detach Faces

The **Detach Faces** ( ) page is used to detach faces from a solid object (the parent) to form a new solid object (the child). Select the faces you want to detach in the Graphics window. They appear in the **Faces to detach** list.

The **Parent heal method** list determines how to replace the detached faces in the parent object: **Fill** means that a new face is constructed, and **Patch** means that the adjacent faces are grown or shrunk to heal the wound.

The **Child heal method** list controls how to construct the child solid from the detached faces: **Fill** means that a new face is formed based on the surrounding edges of each wound, **Patch from child** means that the detached faces grow or shrink to form a solid, and **Patch from parent** means that the parent faces surrounding the detached faces grow or shrink to form a solid together with the detached faces.

When you click the **Detach Selected** button, the program detaches the selected faces and adds a node corresponding to this operation to the geometry branch of the model tree. If you want to modify the performed detach operation, select the added node in the geometry branch. Then edit the node's form that appears in the **Settings** window. Click the **Build Selected** button ( ) to see the result of your edits.

Geometry Features

In this section:

Cap Faces

You can add cap faces to fill holes in a geometry (for example, to make a domain for the void inside a cylinder geometry for simulating fluid flow inside the cylinder) or to partition the geometry. To add cap faces to objects, from the **Geometry** toolbar, **Defeaturing and Repair** menu, select **Cap Faces** ( ).

CAP FACES

Select edges that form loops around the faces you want to create. The edges display in the **Bounding edges** list.

A cap face is created for each loop of edges in the input selection. The cap faces are joined with the original objects. If new closed volumes are created by the cap faces, these are converted to solid domains. The selected edges can contain more than one edge loop, but no two loops can have edges or vertices in common. The selected edges can contain edges from more than one object. In this case, each object is processed individually. This means that two edges or vertices can overlap as long as they are not in the same object. It also means that if new closed volumes are created, but bounded by faces from more than one object, these volumes are not converted to solid domains. If you want to perform a **Cap Faces** operation involving more than one object, first unite the objects using a **Union** operation.

SELECTIONS OF RESULTING ENTITIES

If you want to make the resulting entities contribute to a cumulative selection, select a cumulative selection from the **Contribute to** list (the default, **None**, gives no contribution), or click the **New** button to create a new cumulative selection (see Cumulative Selections in the *COMSOL Multiphysics Reference Manual*).

Select the **Resulting objects selection** check box to create predefined selections (for all levels — objects, domains, boundaries, edges, and points — that are applicable) in subsequent nodes in the geometry sequence. To also make all or one of the types of resulting entities (domains, boundaries, edges, and points) that the resulting objects consist of available as selections in all applicable selection lists (in physics and materials settings, for example), choose an option from the **Show in physics** list: **All levels**, **Domain selection**, **Boundary selection**, **Edge selection**, or **Point selection**. The default is **Domain selection**, which is suitable for use with materials and physics defined in domains. For use with a boundary condition, for example, choose **Boundary selection**. These selections do not appear as separate selection nodes in the model tree. Select **Off** to not make any selection available outside of the geometry sequence.

Delete Faces

To delete and replace faces from an object, from the **Geometry** toolbar, **CAD Defeaturing** menu, select **Delete Faces** ( ). This opens the Delete Faces window. When the deletion operation has been performed, you can modify it by editing the corresponding Delete Faces node that appears in the geometry branch by clicking it.

> The Delete Faces tool can only be applied to objects that are represented by the Parasolid® geometry kernel, also called CAD objects.

DELETE FACES

In the **Faces to delete** list, select the faces you want to delete. In the **Heal method** list, select the method to use for replacing the deleted faces: **Fill** means that the deleted faces are replaced with a new face, while **Patch** means that the adjacent faces are grown or shrunk to heal the wound. Select the **Heal as through hole** check box if you have selected faces that make up a hole that you want to delete.

SELECTIONS OF RESULTING ENTITIES

If you want to make the resulting entities contribute to a cumulative selection, select a cumulative selection from the **Contribute to** list (the default, **None**, gives no contribution), or click the **New** button to create a new cumulative selection (see Cumulative Selections in the *COMSOL Multiphysics Reference Manual*).

Select the **Resulting objects selection** check box to create predefined selections (for all levels — objects, domains, boundaries, edges, and points — that are applicable) in subsequent nodes in the geometry sequence. To also make all or one of the types of resulting entities (domains, boundaries, edges, and points) that the resulting objects consist of available as selections in all applicable selection lists (in physics and materials settings, for example), choose an option from the **Show in physics** list: **All levels**, **Domain selection**, **Boundary selection**, **Edge selection**, or **Point selection**. The default is **Domain selection**, which is suitable for use with materials and physics defined in domains. For use with a boundary condition, for example, choose **Boundary selection**. These selections do not appear as separate selection nodes in the model tree. Select **Off** to not make any selection available outside of the geometry sequence.

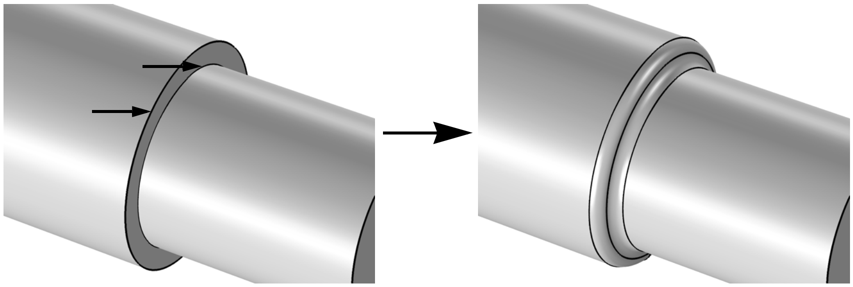
Delete Fillets

To delete fillets from an object, from the **Geometry** toolbar, **Defeaturing and Repair** menu, select **Delete Fillets** ( ). This opens the **Delete Fillets** window, see Finding and Deleting Small Details. When the deletion operation has been performed, you can modify it by editing the corresponding Delete Fillets node that appears in the geometry branch by clicking it.

The Delete Fillets tool can only be applied to objects that are represented by the Parasolid® geometry kernel, also called CAD objects.

Note that fillets found on nonmanifold objects are not possible to delete. An example of a nonmanifold object is an object with several domains. Such an object can for example result from a Union or a Partition operation. To remove the fillets make sure to defeature the geometry objects before applying Boolean operations that result in nonmanifold objects.

The Delete Fillets tool cannot delete fillets for which the adjacent faces cannot be extended to cover the gap. The figure below shows an example of such fillets. Applying the fillets on the highlighted edges deletes the annular face from the geometry, which cannot be recreated if the fillets are to be deleted.



DELETE FILLETS

In the **Input objects** list, select the objects you want to delete fillets from. In the field **Maximum fillet radius**, enter the maximum size of the fillets you want to delete. When you click the **Find Fillets** button, a list of fillets with radius smaller that the given value is shown in the **Fillet selection** list. If **Deletion type** is **All fillets**, all such fillets are deleted. You can delete a subset of these fillets by clicking in the **Fillet selection** list, and choosing **Selected fillets** in the **Deletion type** list.

SELECTIONS OF RESULTING ENTITIES

If you want to make the resulting entities contribute to a cumulative selection, select a cumulative selection from the **Contribute to** list (the default, **None**, gives no contribution), or click the **New** button to create a new cumulative selection (see Cumulative Selections in the *COMSOL Multiphysics Reference Manual*).

Select the **Resulting objects selection** check box to create predefined selections (for all levels — objects, domains, boundaries, edges, and points — that are applicable) in subsequent nodes in the geometry sequence. To also make all or one of the types of resulting entities (domains, boundaries, edges, and points) that the resulting objects consist of available as selections in all applicable selection lists (in physics and materials settings, for example), choose an option from the **Show in physics** list: **All levels**, **Domain selection**, **Boundary selection**, **Edge selection**, or **Point selection**. The default is **Domain selection**, which is suitable for use with materials and physics defined in domains. For use with a boundary condition, for example, choose **Boundary selection**. These selections do not appear as separate selection nodes in the model tree. Select **Off** to not make any selection available outside of the geometry sequence.

Delete Holes

To delete cylindrical holes from an object, from the **Geometry** toolbar, **Defeaturing and Repair** menu, select **Delete Holes** ( ). This opens the **Delete Holes** window, see Finding and Deleting Small Details. When the deletion operation has been performed, you can modify it by editing the corresponding Delete Fillets node that appears in the geometry branch by clicking it.

> The Delete Holes tool can only be applied to objects that are represented by the CAD kernel; see Converting Objects to CAD Kernel Representation.

Note that holes found on nonmanifold objects are not possible to delete. An example of a nonmanifold object is an object with several domains. Such an object can for example result from a Union or a Partition operation. To remove the holes make sure to defeature the geometry objects before applying Boolean operations that result in nonmanifold objects.

DELETE HOLES

In the **Input objects** list, select the objects you want to delete holes from. In the field **Maximum hole radius**, enter the maximum size of the holes you want to delete. When you click the **Find Holes** button, a list of holes with radius smaller that the given value is shown in the **Hole selection** list. If **Deletion type** is **All holes**, all such holes are deleted. You can delete a subset of these holes by clicking in the **Hole selection** list, and choosing **Selected holes** in the **Deletion type** list.

SELECTIONS OF RESULTING ENTITIES

If you want to make the resulting entities contribute to a cumulative selection, select a cumulative selection from the **Contribute to** list (the default, **None**, gives no contribution), or click the **New** button to create a new cumulative selection (see Cumulative Selections in the *COMSOL Multiphysics Reference Manual*).

Select the **Resulting objects selection** check box to create predefined selections (for all levels — objects, domains, boundaries, edges, and points — that are applicable) in subsequent nodes in the geometry sequence. To also make all or one of the types of resulting entities (domains, boundaries, edges, and points) that the resulting objects consist of available as selections in all applicable selection lists (in physics and materials settings, for example), choose an option from the **Show in physics** list: **All levels**, **Domain selection**, **Boundary selection**, **Edge selection**, or **Point selection**. The default is **Domain**

selection, which is suitable for use with materials and physics defined in domains. For use with a boundary condition, for example, choose **Boundary selection**. These selections do not appear as separate selection nodes in the model tree. Select **Off** to not make any selection available outside of the geometry sequence.

Delete Short Edges

To delete short edges from an object, from the **Geometry** toolbar, **Defeaturing and Repair** menu, select **Delete Short Edges** ( ). This opens the **Delete Short Edges** window, see Finding and Deleting Small Details. When the deletion operation has been performed, you can modify it by editing the corresponding **Delete Short Edges** node that appears in the geometry branch by clicking it.

> The Delete Short Edges tool can only be applied to objects that are represented by the Parasolid® geometry kernel, also called CAD objects.

Note that this defeaturing tool cannot find short edges on nonmanifold objects. An example of a nonmanifold object is an object with several domains. Such an object can for example result from a Union or a Partition operation. To avoid this situation defeature the geometry objects before applying Boolean operations that result in nonmanifold objects.

DELETE SHORT EDGES

In the **Input objects** list, select the objects you want to delete short edges from. In the field **Maximum edge length**, enter the maximum length of the edges you want to delete. When you click the **Find Short Edges** button, a list of edges with length smaller that the given value is shown in the **Short edge selection** list. If **Deletion type** is **All short edges**, all such edges are deleted. You can delete a subset of these edges by clicking in the **Short edge selection** list, and choosing **Selected short edges** in the **Deletion type** list.

SELECTIONS OF RESULTING ENTITIES

If you want to make the resulting entities contribute to a cumulative selection, select a cumulative selection from the **Contribute to** list (the default, **None**, gives no contribution), or click the **New** button to create a new cumulative selection (see Cumulative Selections in the *COMSOL Multiphysics Reference Manual*).

Select the **Resulting objects selection** check box to create predefined selections (for all levels — objects, domains, boundaries, edges, and points — that are applicable) in subsequent nodes in the geometry sequence. To also make all or one of the types of resulting entities (domains, boundaries, edges, and points) that the resulting objects consist of available as selections in all applicable selection lists (in physics and materials settings, for example), choose an option from the **Show in physics** list: **All levels**, **Domain selection**, **Boundary selection**, **Edge selection**, or **Point selection**. The default is **Domain selection**, which is suitable for use with materials and physics defined in domains. For use with a boundary condition, for example, choose **Boundary selection**. These selections do not appear as separate selection nodes in the model tree. Select **Off** to not make any selection available outside of the geometry sequence.

Delete Sliver Faces

To delete sliver faces from an object, from the **Geometry** toolbar, **Defeaturing and Repair** menu, select **Delete Sliver Faces** ( ). This opens the **Delete Sliver Faces** window, see Finding and Deleting Small Details. When the deletion operation has been performed, you can modify it by editing the corresponding Delete Sliver Faces node that appears in the geometry branch by clicking it.

> The Delete Sliver Faces tool can only be applied to objects that are represented by the Parasolid® geometry kernel, also called CAD objects.

Note that this defeaturing tool cannot find sliver faces on nonmanifold objects. An example of a nonmanifold object is an object with several domains. Such an object can for example result from a Union or a Partition operation. To avoid this situation defeature the geometry objects before applying Boolean operations that result in nonmanifold objects.

DELETE SLIVER FACES

In the **Input objects** list, select the objects you want to delete sliver faces from. In the field **Maximum face width**, enter the maximum width of the faces you want to delete. When you click the **Find Sliver Faces** button, a list of faces with width smaller that the given value are shown in the **Sliver faces selection** list. If **Deletion type** is **All sliver faces**, all such faces are deleted. You can delete a subset of these faces by clicking in the **Sliver face selection** list, and choosing **Selected sliver faces** in the **Deletion type** list.

SELECTIONS OF RESULTING ENTITIES

If you want to make the resulting entities contribute to a cumulative selection, select a cumulative selection from the **Contribute to** list (the default, **None**, gives no contribution), or click the **New** button to create a new cumulative selection (see Cumulative Selections in the *COMSOL Multiphysics Reference Manual*).

Select the **Resulting objects selection** check box to create predefined selections (for all levels — objects, domains, boundaries, edges, and points — that are applicable) in subsequent nodes in the geometry sequence. To also make all or one of the types of resulting entities (domains, boundaries, edges, and points) that the resulting objects consist of available as selections in all applicable selection lists (in physics and materials settings, for example), choose an option from the **Show in physics** list: **All levels**, **Domain selection**, **Boundary selection**, **Edge selection**, or **Point selection**. The default is **Domain selection**, which is suitable for use with materials and physics defined in domains. For use with a boundary condition, for example, choose **Boundary selection**. These selections do not appear as separate selection nodes in the model tree. Select **Off** to not make any selection available outside of the geometry sequence.

Delete Small Faces

To delete small faces from an object, from the **Geometry** toolbar, **Defeaturing and Repair** menu, select **Delete Small Faces** ( ). This opens the **Delete Small Faces** window, see Finding and Deleting Small Details. When the deletion operation has been performed, you can modify it by editing the corresponding Delete Small Faces node that appears in the geometry branch by clicking it.

> The Delete Small Faces tool can only be applied to objects that are represented by the Parasolid® geometry kernel, also called CAD objects.

Note that this defeaturing tool cannot find small faces on nonmanifold objects. An example of a nonmanifold object is an object with several domains. Such an object can for example result from a Union or a Partition operation. To avoid this situation defeature the geometry objects before applying Boolean operations that result in nonmanifold objects.

DELETE SMALL FACES

In the **Input objects** list, select the objects you want to delete small faces from. In the field **Maximum face size**, enter the maximum diameter of the faces you want to delete. When you click the **Find Small Faces** button, a list of faces with diameter smaller that the given value appears in the **Small faces selection** list. If **Deletion type** is **All small faces**, all such faces are deleted. You can delete a subset of these faces by clicking in the **Small face selection** list, and choosing **Selected small faces** in the **Deletion type** list.

SELECTIONS OF RESULTING ENTITIES

If you want to make the resulting entities contribute to a cumulative selection, select a cumulative selection from the **Contribute to** list (the default, **None**, gives no contribution), or click the **New** button to create a new cumulative selection (see Cumulative Selections in the *COMSOL Multiphysics Reference Manual*).

Select the **Resulting objects selection** check box to create predefined selections (for all levels — objects, domains, boundaries, edges, and points — that are applicable) in subsequent nodes in the geometry sequence. To also make all or one of the types of resulting entities (domains, boundaries, edges, and points) that the resulting objects consist of available as selections in all applicable selection lists (in physics and materials settings, for example), choose an option from the **Show in physics** list: **All levels**, **Domain selection**, **Boundary selection**, **Edge selection**, or **Point selection**. The default is **Domain selection**, which is suitable for use with materials and physics defined in domains. For use with a boundary condition, for example, choose **Boundary selection**. These selections do not appear as separate selection nodes in the model tree. Select **Off** to not make any selection available outside of the geometry sequence.

Delete Spikes

A spike is a long and narrow protrusion on an edge or corner of a face defined by two or three edges. To delete spikes from an object, from the **Geometry** toolbar, **Defeaturing and Repair** menu, select **Delete Spikes** ( ). This opens the **Delete Spikes** window, see Finding and Deleting Small Details. When the deletion operation has been performed, you can modify it by editing the corresponding **Delete Spikes** node that appears in the geometry branch by clicking it.

> The Delete Spikes tool can only be applied to objects that are represented by the Parasolid® geometry kernel, also called CAD objects.

Note that this defeaturing tool cannot find spikes on faces that belong to nonmanifold objects. An example of a nonmanifold object is an object with several domains. Such an object can for example result from a Union or a Partition operation. To avoid this situation defeature the geometry objects before applying Boolean operations that result in nonmanifold objects.

DELETE SPIKES

In the **Input objects** list, select the objects you want to delete spikes from. In the field **Maximum spike width**, enter the maximum width of the spikes you want to delete.

When you click the **Find Spikes** button, a list of spikes with width smaller that the given value are shown in the **Spike selection** list. If **Deletion type** is **All spikes**, all such spikes are deleted. You can delete a subset of these spikes by clicking in the **Spike selection** list, and choosing **Selected spikes** in the **Deletion type** list.

SELECTIONS OF RESULTING ENTITIES

If you want to make the resulting entities contribute to a cumulative selection, select a cumulative selection from the **Contribute to** list (the default, **None**, gives no contribution), or click the **New** button to create a new cumulative selection (see Cumulative Selections in the *COMSOL Multiphysics Reference Manual*).

Select the **Resulting objects selection** check box to create predefined selections (for all levels — objects, domains, boundaries, edges, and points — that are applicable) in subsequent nodes in the geometry sequence. To also make all or one of the types of resulting entities (domains, boundaries, edges, and points) that the resulting objects consist of available as selections in all applicable selection lists (in physics and materials settings, for example), choose an option from the **Show in physics** list: **All levels**, **Domain selection**, **Boundary selection**, **Edge selection**, or **Point selection**. The default is **Domain selection**, which is suitable for use with materials and physics defined in domains. For use with a boundary condition, for example, choose **Boundary selection**. These selections do not appear as separate selection nodes in the model tree. Select **Off** to not make any selection available outside of the geometry sequence.

Detach Faces

To detach faces from an object (the parent) and form a new object (the child), from the **Geometry** toolbar, **Defeaturing and Repair** menu, select **Detach Faces** (). This opens the Detach Faces window. When the detach operation has been performed, you can modify it by editing the corresponding **Detach Faces** node that appears in the geometry branch by clicking it.

The Detach Faces tool can only be applied to objects that are represented by the Parasolid® geometry kernel, also called CAD objects.

DETACH FACES

Select the faces you want to detach in the **Graphics** window. They appear in the **Faces to detach** list.

The **Parent heal method** list determines how to replace the detached faces in the parent object: **Fill** means that a new face is constructed, and **Patch** means that the adjacent faces grow or shrink to heal the wound.

The **Child heal method** list controls how to construct the child solid from the detached faces: **Fill** means that a new face is formed based on the surrounding edges of each wound, **Patch from child** means that the detached faces are grown or shrunk to form a solid, and **Patch from parent** means that the parent faces surrounding the detached faces are grown or shrunk to form a solid together with the detached faces.

SELECTIONS OF RESULTING ENTITIES

If you want to make the resulting entities contribute to a cumulative selection, select a cumulative selection from the **Contribute to** list (the default, **None**, gives no contribution), or click the **New** button to create a new cumulative selection (see Cumulative Selections in the *COMSOL Multiphysics Reference Manual*).

Select the **Resulting objects selection** check box to create predefined selections (for all levels — objects, domains, boundaries, edges, and points — that are applicable) in subsequent nodes in the geometry sequence. To also make all or one of the types of resulting entities (domains, boundaries, edges, and points) that the resulting objects consist of available as selections in all applicable selection lists (in physics and materials settings, for example), choose an option from the **Show in physics** list: **All levels**, **Domain selection**, **Boundary selection**, **Edge selection**, or **Point selection**. The default is **Domain selection**, which is suitable for use with materials and physics defined in domains. For use with a boundary condition, for example, choose **Boundary selection**. These selections do not appear as separate selection nodes in the model tree. Select **Off** to not make any selection available outside of the geometry sequence.

Knit to Solid

To knit surface objects to form solid objects, from the **Geometry** toolbar, **Defeaturing and Repair** menu, select **Knit to Solid** ( ).

KNIT TO SOLID

Select the objects to knit together in the **Graphics** window. They appear in the **Input objects** list.

The knitting merges edges that have a distance smaller that the **Absolute repair tolerance** and deletes gaps and spikes smaller than the **Absolute repair tolerance**. If the

Fill holes check box is selected the operation attempts to generate new faces to replace missing geometry.

SELECTIONS OF RESULTING ENTITIES

If you want to make the resulting entities contribute to a cumulative selection, select a cumulative selection from the **Contribute to** list (the default, **None**, gives no contribution), or click the **New** button to create a new cumulative selection (see Cumulative Selections in the *COMSOL Multiphysics Reference Manual*).

Select the **Resulting objects selection** check box to create predefined selections (for all levels — objects, domains, boundaries, edges, and points — that are applicable) in subsequent nodes in the geometry sequence. To also make all or one of the types of resulting entities (domains, boundaries, edges, and points) that the resulting objects consist of available as selections in all applicable selection lists (in physics and materials settings, for example), choose an option from the **Show in physics** list: **All levels**, **Domain selection**, **Boundary selection**, **Edge selection**, or **Point selection**. The default is **Domain selection**, which is suitable for use with materials and physics defined in domains. For use with a boundary condition, for example, choose **Boundary selection**. These selections do not appear as separate selection nodes in the model tree. Select **Off** to not make any selection available outside of the geometry sequence.

Repair

To repair objects, from the **Geometry** toolbar, **Defeaturing and Repair** menu, select **Repair** ( ).

REPAIR

Select the objects to repair in the **Graphics** window. They appear in the **Input objects** list.

The software tries to repair defects and remove details smaller than the **Absolute import tolerance**. More precisely:

- Entities with invalid sense
- Invalid edge and vertex tolerances
- Invalid manifolds
- Self-intersecting manifolds
- Non-G1 manifolds
- Missing edge or vertex manifolds
- Missing vertex

- Vertices not on curve of edge
- Edges and vertices not on surface of face
- Removal of surface self-intersections that lie outside the face
- Splitting at edge intersections which have no vertex
- Removal of discontinuities by either splitting or smoothing
- Remove small features (short edges, small faces, sliver faces, and spikes)

Select the option **Simplify curves and surfaces** to also simplify the underlying curve and surface manifolds of the geometric entities. Repairing objects with this option may help in some cases when Boolean operations on the objects fail.

SELECTIONS OF RESULTING ENTITIES

If you want to make the resulting entities contribute to a cumulative selection, select a cumulative selection from the **Contribute to** list (the default, **None**, gives no contribution), or click the **New** button to create a new cumulative selection (see Cumulative Selections in the *COMSOL Multiphysics Reference Manual*).

Select the **Resulting objects selection** check box to create predefined selections (for all levels — objects, domains, boundaries, edges, and points — that are applicable) in subsequent nodes in the geometry sequence. To also make all or one of the types of resulting entities (domains, boundaries, edges, and points) that the resulting objects consist of available as selections in all applicable selection lists (in physics and materials settings, for example), choose an option from the **Show in physics** list: **All levels**, **Domain selection**, **Boundary selection**, **Edge selection**, or **Point selection**. The default is **Domain selection**, which is suitable for use with materials and physics defined in domains. For use with a boundary condition, for example, choose **Boundary selection**. These selections do not appear as separate selection nodes in the model tree. Select **Off** to not make any selection available outside of the geometry sequence.

4

Programming and Command Reference

In this section you find detailed COMSOL® API reference information for the geometry features in LiveLink™ *for* Revit®.

Defeaturing Tools

To remove unnecessary details in objects imported from a 3D CAD file, you can use the defeaturing tools. You access these by typing:

```
model.component(<ctag>).geom(<tag>).defeaturing("Fillets");
model.component(<ctag>).geom(<tag>).defeaturing("Holes");
model.component(<ctag>).geom(<tag>).defeaturing("ShortEdges");
model.component(<ctag>).geom(<tag>).defeaturing("SliverFaces");
model.component(<ctag>).geom(<tag>).defeaturing("SmallFaces");
model.component(<ctag>).geom(<tag>).defeaturing("Spikes");
model.component(<ctag>).geom(<tag>).defeaturing("DeleteFaces");
model.component(<ctag>).geom(<tag>).defeaturing("DetachFaces");
```

Using the defeaturing tools you can search for small details, without altering your geometry. If you find small details that you want to remove, a defeaturing tool can create a feature that removes the details from the geometry.

The features corresponding to the defeaturing tools are `DeleteFillets`, `DeleteHoles`, `DeleteShortEdges`, `DeleteSliverFaces`, `DeleteSmallFaces`, `DeleteSpikes`, `DeleteFaces`, and `DetachFaces`. If you already know which details you need to remove, it is also possible to create these features directly using the standard create syntax.

This section includes these topics:

- Defeaturing Tools — Finding and Deleting Small Details
- Defeaturing Tools — Delete Faces
- Defeaturing Tools — Detach Faces

Defeaturing Tools — Finding and Deleting Small Details

The defeaturing tools `Fillets`, `Holes`, `ShortEdges`, `SliverFaces`, `SmallFaces`, and `Spikes` search for and delete details smaller than a given size. First select the objects you want to examine by typing, for example,

```
model.component(<ctag>).geom(<tag>).defeaturing("Fillets").
      selection("input").set(<onames>);
```

where *<onames>* is a string array contains the object names.

Set the maximum size of the details (fillets in this case) you want to remove by typing

```
model.component(<ctag>).geom(<tag>).defeaturing("Fillets").
      set("entsize",size);
```

To find the details that are smaller than the given size, type

```
model.component(<ctag>).geom(<tag>).defeaturing("Fillets").
      find();
```

The found details appear in the selection

```
model.component(<ctag>).geom(<tag>).defeaturing("Fillets").
      detail();
```

To get the number of found details, type

```
int nd = model.component(<ctag>).geom(<tag>).
         defeaturing("Fillets").detail().size();
```

To get the names of the found details, type

```
String[] filletNames = model.component(<ctag>).geom(<tag>).
         defeaturing("Fillets").detail().groupNames();
```

In general, a detail (fillet in this case) consists of a number of geometric entities. For example, a fillet consists of a number of faces. To get the entity numbers in the nth detail, type

```
int[] entities = model.component(<ctag>).geom(<tag>).
               defeaturing("Fillets").detail().groupEntities(n);
```

To get the object that contains the nth detail, type

```
String oname = model.component(<ctag>).geom(<tag>).
               defeaturing("Fillets").detail().groupObject(n);
```

To delete all details found, type

```
model.component(<ctag>).geom(<tag>).defeaturing("Fillets").
      deleteAll(<ftag>);
```

This adds a feature, tagged <ftag>, that performs the deletion operation to the geometry sequence, after the current feature, and build this feature. In this case, it adds a DeleteFillets feature.

To delete a subset of the details found, type, for example

```
model.component(<ctag>).geom(<tag>).defeaturing("Fillets").
      detail().setGroup(2,5);
```

to delete fillets number 2 and 5. You can also use, for example,

```
model.component(<ctag>).geom(<tag>).defeaturing("Fillets").
      detail().addGroup(7,8);
```

```
model.component(<ctag>).geom(<tag>).defeaturing("Fillets").
      detail().removeGroup(3);
```

to add and remove details from the selection. Perform the deletion by typing

```
model.component(<ctag>).geom(<tag>).defeaturing("Fillets").
      delete(<ftag>);
```

This adds a `DeleteFillets` feature tagged *<ftag>* after the current feature in the geometry sequence.

DEFEATURING METHODS

`model.component(<ctag>).geom(<tag>).feature(<ftag>).find()` searches for small details, for a defeaturing feature *<ftag>*.

`model.component(<ctag>).geom(<tag>).defeaturing(tooltag).find()` searches for small details, for a defeaturing tool *tooltag*.

`model.component(<ctag>).geom(<tag>).defeaturing(tooltag).detail().selMethod` manipulates the selection of details to remove, for a defeaturing tool *tooltag*.

`model.component(<ctag>).geom(<tag>).feature(<ftag>).detail().selMethod` manipulates the selection of details to remove, for a defeaturing feature *<ftag>*.

`model.component(<ctag>).geom(<tag>).defeaturing(tooltag).delete(<ftag>)` creates a defeaturing feature of type *tooltag*, tagged *<ftag>*, with the properties currently specified in the defeaturing tool. The property `delete` of the created feature is set to `selected`. If the feature *<ftag>* can be built, it is inserted in the geometry sequence after the current feature, otherwise the feature is discarded.

`model.component(<ctag>).geom(<tag>).defeaturing(tooltag).deleteAll(<ftag>)` creates a defeaturing feature of type *tooltag*, tagged *<ftag>*, with the properties currently specified in the defeaturing tool. The property `delete` of the created feature is set to `all`. If the feature *<ftag>* can be built, it is inserted in the geometry sequence after the current feature, otherwise the feature is discarded.

DEFEATURING SELECTION METHODS

For a defeaturing selection sel the following methods are available, in addition to the methods available for a general geometry selection.

> Geometry Object Selection Methods in the *COMSOL Multiphysics Programming Reference Manual*

The find method on the corresponding feature or defeaturing tool provides the defeaturing selection with a list of details. Each detail is a group of geometric entities. Group numbers, *<groups>*, is an array of integers that index into the list of details.

You can select groups either by explicitly referring to group numbers, or by selecting geometric entities. In the latter case, any group that has non-empty intersection with the provided entity selection is selected.

int[] sel.group(*<groups>*) returns the group numbers for the selected groups.

sel.addGroup(*<groups>*) adds the specified groups to the selection.

sel.setGroup(*<groups>*) sets the selection groups.

sel.removeGroup(*<groups>*) removes the specified groups from the selection.

String[] sel.groupNames() returns a list of names of the groups found.

String sel.groupObject(*<group>*) returns the name of the geometry object that contains the specified detail group.

int[] sel.groupEntities(*<group>*) returns the entity numbers of the specified detail group.

int sel.size() returns the number of detail groups found.

Defeaturing Tools — Delete Faces

Use the DeleteFaces tool to delete faces and replace them either with a new face or by growing or shrinking the adjacent faces. Select the faces to delete and properties for the operation like in the corresponding feature DeleteFaces. The deletion is performed when you issue the command

```
model.component(<ctag>).geom(<tag>).defeaturing("DeleteFaces").
      delete(<ftag>);
```

This adds a `DeleteFaces` feature tagged *<ftag>* after the current feature in the geometry sequence.

Defeaturing Tools — Detach Faces

Use the `DetachFaces` tool to detach faces from a solid object (the parent) to form a new solid object (the child). Select the faces to detach and properties for the operation like in the corresponding feature `DetachFaces`. The detach operation is performed when you issue the command

```
model.component(<ctag>).geom(<tag>).defeaturing("DetachFaces").
      delete(<ftag>);
```

Summary of Commands

- CapFaces
- ConvertToCOMSOL
- DeleteFaces
- DeleteFillets
- DeleteHoles
- DeleteShortEdges
- DeleteSliverFaces
- DeleteSmallFaces
- DeleteSpikes
- DetachFaces
- Export, ExportFinal
- Import
- Knit
- LiveLinkRevit
- Repair

Commands Grouped by Function

Commands for Defeaturing

FUNCTION	PURPOSE
`DeleteFaces`	Delete faces from CAD objects and heal the wounds
`DeleteFillets`	Find and delete fillets in CAD objects
`DeleteHoles`	Find and delete holes in CAD objects
`DeleteShortEdges`	Find and delete short edges in CAD objects
`DeleteSliverFaces`	Find and delete sliver faces in CAD objects
`DeleteSmallFaces`	Find and delete small faces in CAD objects
`DeleteSpikes`	Find and delete spikes in CAD objects
`Export, ExportFinal`	Detach faces from CAD objects to form a new solid

Commands for File Import, Export, Conversion and Repair

FUNCTION	PURPOSE
`ConvertToCOMSOL`	Convert CAD Import Module geometry objects to COMSOL objects
`Export, ExportFinal`	Export geometry objects to a 3D CAD file
`Import`	Import geometry objects from a 3D CAD file
`Knit`	Knit surface CAD objects to form solids or surface objects
`Repair`	Repair CAD objects

Commands for Geometry Creation and Modification

FUNCTION	PURPOSE
`CapFaces`	Add cap faces to fill holes in CAD geometries

Commands for Interfacing CAD Software

FUNCTION	PURPOSE
LiveLinkRevit	Synchronize geometry objects with a Revit project

Commands in Alphabetical Order

CapFaces

PURPOSE

Add cap faces to objects.

SYNTAX

```
model.component(<ctag>).geom(<tag>).feature().
      create(<ftag>,"CapFaces");
model.component(<ctag>).geom(<tag>).feature(<ftag>).
      selection(property);
```

DESCRIPTION

```
model.component(<ctag>).geom(<tag>).feature().
      create(<ftag>,"CapFaces")
```

creates a `CapFaces` feature. A cap face is created for each loop of edges in the input selection. The cap faces are joined with the original objects. If new domains are created by the cap faces, these domains are made solid.

The input selection can contain more than one edge loop, but no two loops can have edges or vertices in common.

The input selection can contain edges from more than one object. In this case, each object is processed individually.

TABLE 4-1: AVAILABLE PROPERTIES

PROPERTY	VALUE	DEFAULT	DESCRIPTION
`input`	Selection		The input edges.
`selresult`	`on` | `off`	`off`	Create selections of all resulting objects.
`selresultshow`	`all` | `obj` | `dom` | `bnd` | `edg` | `pnt` | `off`	`dom`	Show selections of resulting objects in physics, materials, and so on, or in part instances. `obj` is not available in a component's geometry. `dom`, `bnd`, and `edg` are not available in all features.
`contributeto`	String	`none`	Tag of cumulative selection to contribute to.

ConvertToCOMSOL

Convert CAD Import Module geometry objects to COMSOL objects.

SYNTAX

```
model.component(<ctag>).geom(<tag>).feature().
      create(<ftag>,"ConvertToCOMSOL");
model.component(<ctag>).geom(<tag>).feature(<ftag>).
      selection(property);
```

DESCRIPTION

```
model.component(<ctag>).geom(<tag>).feature().
      create(<ftag>,"ConvertToCOMSOL")
```

creates a `ConvertToCOMSOL` feature.

TABLE 4-2: AVAILABLE PROPERTIES

PROPERTY	VALUE	DEFAULT	DESCRIPTION
`input`	Selection		Names of input objects.
`selresult`	`on` | `off`	`off`	Create selections of all resulting objects.
`selresultshow`	`all` | `obj` | `dom` | `bnd` | `edg` | `pnt` | `off`	`dom`	Show selections of resulting objects in physics, materials, and so on, or in part instances. `obj` is not available in a component's geometry. `dom`, `bnd`, and `edg` are not available in all features.
`contributeto`	String	`none`	Tag of cumulative selection to contribute to.

SEE ALSO

`Import`

DeleteFaces

Delete faces from CAD objects and heal the wounds.

SYNTAX

```
model.component(<ctag>).geom(<tag>).feature().
      create(<ftag>,"DeleteFaces");
model.component(<ctag>).geom(<tag>).feature(<ftag>).
      selection(property);
model.component(<ctag>).geom(<tag>).feature(<ftag>).
      set(property,<value>);
model.component(<ctag>).geom(<tag>).feature(<ftag>).
      getType(property);

model.component(<ctag>).geom(<tag>).defeaturing("DeleteFaces").
      selection(property)
model.component(<ctag>).geom(<tag>).defeaturing("DeleteFaces").
      set(property,<value>);
model.component(<ctag>).geom(<tag>).defeaturing("DeleteFaces").dele
  te(<ftag>);
```

DESCRIPTION

model.component(*<ctag>*).geom(*<tag>*).defeaturing("DeleteFaces").delete(*<ftag>*) creates a DeleteFaces feature tagged *<ftag>* with the specified properties. If the feature can be built, it is inserted in the geometry sequence after the current feature; otherwise, the feature is discarded.

It is also possible to create a DeleteFaces feature using the standard create method.

TABLE 4-3: AVAILABLE PROPERTIES

<table>
<tr><th>PROPERTY</th><th>VALUE</th><th>DEFAULT</th><th>DESCRIPTION</th></tr>
<tr><td>input</td><td>Selection</td><td></td><td>Faces to delete.</td></tr>
<tr><td>heal</td><td>fill | patch</td><td>patch</td><td>Healing method.</td></tr>
<tr><td>throughhole</td><td>on | off</td><td>off</td><td>Heal as if the removed faces are a through hole.</td></tr>
<tr><td>selresult</td><td>on | off</td><td>off</td><td>Create selections of all resulting objects.</td></tr>
<tr><td>selresultshow</td><td>all | obj | dom | bnd | edg | pnt | off</td><td>dom</td><td>Show selections of resulting objects in physics, materials, and so on, or in part instances. obj is not available in a component's geometry. dom, bnd, and edg are not available in all features.</td></tr>
<tr><td>contributeto</td><td>String</td><td>none</td><td>Tag of cumulative selection to contribute to.</td></tr>
</table>

The faces in the property input are deleted from their objects. The resulting object is healed so that a solid object is obtained. If heal is fill, a new face is formed based on the surrounding edges of each wound. If heal is `patch`, the surrounding faces of each wound are grown or shrunk to heal the wound.

When you delete through holes, set the `throughhole` property to `on` to indicate that the two wounds from where the hole entered and exited the geometry are to be healed independently instead of as a single wound. If `throughhole` is `off`, the wound would be healed with a single new face that would just recreate the hole.

EXAMPLE

The following example imports the file `defeaturing_demo_2.mphbin`, and removes a hole from the geometry model.

```
Model model = ModelUtil.create("Model1");
model.component.create("comp1");
model.component("comp1").geom().create("geom1",3);
model.component("comp1").geom("geom1").feature().
      create("imp1","Import");
model.component("comp1").geom("geom1").feature("imp1").
      set("filename","defeaturing_demo_2.mphbin");
model.component("comp1").geom("geom1").run("imp1");
model.component("comp1").geom("geom1").feature().
      create("dfa1","DeleteFaces");
model.component("comp1").geom("geom1").feature("dfa1").
      selection("input").set("imp1",6,7,8,9,11,12,13);
model.component("comp1").geom("geom1").run();
```

COMPATIBILITY

The following property is no longer supported:

TABLE 4-4: OBSOLETE PROPERTIES

PROPERTY	VALUE	DEFAULT	DESCRIPTION
`Out`	`stx` | `ftx` | `ctx` | `ptx`	none	Output variables

SEE ALSO

`DeleteFillets`, `DeleteSliverFaces`, `DeleteSmallFaces`, `Export`, `ExportFinal`

DeleteFillets

Find and delete fillets in CAD objects.

SYNTAX

```
model.geom(<tag>).feature().create(<ftag>,"DeleteFillets");
model.geom(<tag>).feature(<ftag>).selection(property);
model.geom(<tag>).feature(<ftag>).set(property,<value>);
model.geom(<tag>).feature(<ftag>).getType(property);
model.geom(<tag>).feature(<ftag>).find();
model.geom(<tag>).feature(<ftag>).detail();

model.geom(<tag>).defeaturing("Fillets").selection(property);
model.geom(<tag>).defeaturing("Fillets").set(property,<value>);
model.geom(<tag>).defeaturing("Fillets").find();
model.geom(<tag>).defeaturing("Fillets").detail();
model.geom(<tag>).defeaturing("Fillets").delete(<ftag>);
model.geom(<tag>).defeaturing("Fillets").deleteAll(<ftag>);
```

DESCRIPTION

`model.geom(<tag>).defeaturing("Fillets").delete(<ftag>)` creates a `DeleteFillets` feature tagged *<ftag>* with the specified properties. The property `delete` is set to `selected`. If the feature can be built, it is inserted in the geometry sequence after the current feature; otherwise, the feature is discarded.

`model.geom(<tag>).defeaturing("Fillets").deleteAll(<ftag>)` works as the `delete` method, but the property `delete` is set to `all`.

It is also possible to create the `DeleteFillets` feature using the standard `create` method. The following properties are available..

TABLE 4-5: AVAILABLE PROPERTIES

PROPERTY	VALUE	DEFAULT	DESCRIPTION
`delete`	`all` | `selected`	`selected`	Delete all fillets of given size, or a selection. Only available for the feature.
`entsize`	`double`	`1e-3`	Maximum fillet radius.
`input`	`Selection`		Names of input objects.
`selresult`	`on` | `off`	`off`	Create selections of all resulting objects.

TABLE 4-5: AVAILABLE PROPERTIES

PROPERTY	VALUE	DEFAULT	DESCRIPTION
`selresultshow`	`all` | `obj` | `dom` | `bnd` | `edg` | `pnt` | `off`	`dom`	Show selections of resulting objects in physics, materials, and so on, or in part instances. `obj` is not available in a component's geometry. `dom`, `bnd`, and `edg` are not available in all features.
`contributeto`	String	`none`	Tag of cumulative selection to contribute to.

`model.geom(<tag>).feature(<ftag>).find()` searches the input objects for fillets with radius less than `entsize`.

`model.geom(<tag>).feature(<ftag>).detail()` returns a selection object where you can select a subset of the fillets found.

The `find` and `detail` methods of `model.geom(<tag>).defeaturing("Fillets")` has the corresponding functionality for the defeaturing tool.

Only faces that can be deleted without invalidating the object are deleted. If a fillet was not possible to delete, a warning is given, accessible through `model.geom(<tag>).feature(<ftag>).problem()`.

COMPATIBILITY

The following property is no longer supported:

TABLE 4-6: OBSOLETE PROPERTIES

PROPERTY	VALUE	DEFAULT	DESCRIPTION
`Out`	`stx` | `ftx` | `ctx` | `ptx`	none	Output variables

EXAMPLE

The following example imports the CAD object in the COMSOL Multiphysics geometry file `defeaturing_demo_3.mphbin` and finds all fillets with radius less than $4 \cdot 10^{-3}$. The first of these fillets is deleted.

```
Model model = ModelUtil.create("Model1");
model.geom().create("geom1",3);
model.geom("geom1").feature().create("imp1","Import");
model.geom("geom1").feature("imp1").set("filename",
      "defeaturing_demo_3.mphbin");
model.geom("geom1").run("imp1");
model.geom("geom1").feature().create("dfi1","DeleteFillets");
model.geom("geom1").feature("dfi1").selection("input").
```

```
    set("imp1");
model.geom("geom1").feature("dfi1").set("entsize",4e-3);
model.geom("geom1").feature("dfi1").find();
model.geom("geom1").feature("dfi1").detail().setGroup(1);
model.geom("geom1").run();
```

SEE ALSO

DeleteFaces

DeleteHoles

Find and delete holes in CAD objects.

SYNTAX

```
model.geom(<tag>).feature().create(<ftag>,"DeleteHoles");
model.geom(<tag>).feature(<ftag>).selection(property);
model.geom(<tag>).feature(<ftag>).set(property,<value>);
model.geom(<tag>).feature(<ftag>).getType(property);
model.geom(<tag>).feature(<ftag>).find();
model.geom(<tag>).feature(<ftag>).detail();

model.geom(<tag>).defeaturing("Holes").selection(property);
model.geom(<tag>).defeaturing("Holes").set(property,<value>);
model.geom(<tag>).defeaturing("Holes").find();
model.geom(<tag>).defeaturing("Holes").detail();
model.geom(<tag>).defeaturing("Holes").delete(<ftag>);
model.geom(<tag>).defeaturing("Holes").deleteAll(<ftag>);
```

DESCRIPTION

`model.geom(<tag>).defeaturing("Holes").delete(<ftag>)` creates a `DeleteHoles` feature tagged *<ftag>* with the specified properties. The property `delete` is set to `selected`. If the feature can be built, it is inserted in the geometry sequence after the current feature; otherwise, the feature is discarded.

`model.geom(<tag>).defeaturing("Holes").deleteAll(<ftag>)` works as the `delete` method, but the property `delete` is set to `all`.

It is also possible to create the `DeleteHoles` feature using the standard `create` method. The following properties are available..

TABLE 4-7: AVAILABLE PROPERTIES

PROPERTY	VALUE	DEFAULT	DESCRIPTION
`delete`	`all` | `selected`	`selected`	Delete all holes of given size, or a selection. Only available for the feature.
`entsize`	`double`	`1e-3`	Maximum hole radius.
`input`	`Selection`		Names of input objects.
`selresult`	`on` | `off`	`off`	Create selections of all resulting objects.
`selresultshow`	`all` | `obj` | `dom` | `bnd` | `edg` | `pnt` | `off`	`dom`	Show selections of resulting objects in physics, materials, and so on, or in part instances. `obj` is not available in a component's geometry. `dom`, `bnd`, and `edg` are not available in all features.
`contributeto`	String	`none`	Tag of cumulative selection to contribute to.

`model.geom(`*<tag>*`).feature(`*<ftag>*`).find()` searches the input objects for holes with radius less than `entsize`.

`model.geom(`*<tag>*`).feature(`*<ftag>*`).detail()` returns a selection object where you can select a subset of the holes found.

The `find` and `detail` methods of `model.geom(`*<tag>*`).defeaturing("Holes")` has the corresponding functionality for the defeaturing tool.

Only faces that can be deleted without invalidating the object are deleted. If a hole was not possible to delete, a warning is given, accessible through `model.geom(`*<tag>*`).feature(`*<ftag>*`).problem()`.

COMPATIBILITY

The following property is no longer supported:

TABLE 4-8: OBSOLETE PROPERTIES

PROPERTY	VALUE	DEFAULT	DESCRIPTION
`Out`	`stx` | `ftx` | `ctx` | `ptx`	none	Output variables

EXAMPLE

The following example imports the CAD object in the COMSOL Multiphysics geometry file `defeaturing_demo_3.mphbin` and finds all holes with radius less than $4{\cdot}10^{-2}$. The first four of these holes are deleted.

```
Model model = ModelUtil.create("Model1");
model.geom().create("geom1",3);
model.geom("geom1").feature().create("imp1","Import");
model.geom("geom1").feature("imp1").set("filename",
      "defeaturing_demo_3.mphbin");
model.geom("geom1").run("imp1");
model.geom("geom1").feature().create("dho1","DeleteHoles");
model.geom("geom1").feature("dho1").selection("input").
      set("imp1");
model.geom("geom1").feature("dho1").set("entsize",4e-2);
model.geom("geom1").feature("dho1").find();
model.geom("geom1").feature("dho1").detail().setGroup(1, 2, 3, 4);
model.geom("geom1").run();
```

SEE ALSO

`DeleteFaces`

DeleteShortEdges

Find and delete short edges in CAD objects.

SYNTAX

```
model.geom(<tag>).feature().create(<ftag>,"DeleteShortEdges");
model.geom(<tag>).feature(<ftag>).selection(property);
model.geom(<tag>).feature(<ftag>).set(property,<value>);
model.geom(<tag>).feature(<ftag>).getType(property);
model.geom(<tag>).feature(<ftag>).find();
model.geom(<tag>).feature(<ftag>).detail();

model.geom(<tag>).defeaturing("ShortEdges").selection(property);
model.geom(<tag>).defeaturing("ShortEdges").
      set(property,<value>);
model.geom(<tag>).defeaturing("ShortEdges").find();
model.geom(<tag>).defeaturing("ShortEdges").detail();
model.geom(<tag>).defeaturing("ShortEdges").delete(<ftag>);
model.geom(<tag>).defeaturing("ShortEdges").deleteAll(<ftag>);
```

DESCRIPTION

`model.geom(<tag>).defeaturing("ShortEdges").delete(<ftag>)` creates a `DeleteShortEdges` feature tagged *<ftag>* with the specified properties. The property `delete` is set to `selected`. If the feature can be built, it is inserted in the geometry sequence after the current feature; otherwise, the feature is discarded.

`model.geom(`*<tag>*`).defeaturing("ShortEdges").deleteAll(`*<ftag>*`)` works as the `delete` method, but the property `delete` is set to `all`.

It is also possible to create a `DeleteShortEdges` feature using the standard `create` method. The following properties are available.

TABLE 4-9: AVAILABLE PROPERTIES

PROPERTY	VALUE	DEFAULT	DESCRIPTION
`delete`	`all` | `selected`	`selected`	Delete all edges of given size, or a selection. Only available for the feature.
`entsize`	`double`	`1e-3`	Maximum edge length
`input`	`Selection`		Names of input objects
`selresult`	`on` | `off`	`off`	Create selections of all resulting objects.
`selresultshow`	`all` | `obj` | `dom` | `bnd` | `edg` | `pnt` | `off`	`dom`	Show selections of resulting objects in physics, materials, and so on, or in part instances. `obj` is not available in a component's geometry. `dom`, `bnd`, and `edg` are not available in all features.
`contributeto`	String	`none`	Tag of cumulative selection to contribute to.

`model.geom(`*<tag>*`).feature(`*<ftag>*`).find()` searches the input objects for edges of length less than `entsize`.

`model.geom(`*<tag>*`).feature(`*<ftag>*`).detail()` returns a selection object where you can select a subset of the edge sets found.

The `find` and `detail` methods of `model.geom(`*<tag>*`).defeaturing("ShortEdges")` has the corresponding functionality for the defeaturing tool.

Only edges that can be deleted without invalidating the object are deleted. If an edge was not possible to delete, a warning is given, accessible through `model.geom(`*<tag>*`).feature(`*<ftag>*`).problem()`.

COMPATIBILITY

The lengths of the edges are no longer returned.

The following property is no longer supported:

TABLE 4-10: OBSOLETE PROPERTIES

PROPERTY	VALUE	DEFAULT	DESCRIPTION
`Out`	`stx` | `ftx` | `ctx` | `ptx`	none	Output variables

EXAMPLE

The following example imports the file `defeaturing_demo_4.x_b` and finds all edges with length less than $3 \cdot 10^{-3}$. The first of these edges is deleted.

```
Model model = ModelUtil.create("Model1");
model.geom().create("geom1",3);
model.geom("geom1").feature().create("imp1","Import");
model.geom("geom1").feature("imp1").set("filename",
     "defeaturing_demo_4.x_b");
model.geom("geom1").runAll();
model.geom("geom1").feature().
      create("dse1","DeleteShortEdges");
model.geom("geom1").feature("dse1").selection("input").
      set("imp1");
model.geom("geom1").feature("dse1").set("entsize",3e-3);
model.geom("geom1").feature("dse1").find();
model.geom("geom1").feature("dse1").detail().setGroup(1);
model.geom("geom1").runAll();
```

DeleteSliverFaces

Find and delete sliver faces in CAD objects.

SYNTAX

```
model.geom(gname).feature().create(<ftag>,"DeleteSliverFaces");
model.geom(<tag>).feature(<ftag>).selection(property);
model.geom(<tag>).feature(<ftag>).set(property,<value>);
model.geom(<tag>).feature(<ftag>).getType(property);
model.geom(<tag>).feature(<ftag>).find();
model.geom(<tag>).feature(<ftag>).detail();

model.geom(<tag>).defeaturing("SliverFaces").selection(property);
model.geom(<tag>).defeaturing("SliverFaces").
      set(property,<value>);
model.geom(<tag>).defeaturing("SliverFaces").find();
model.geom(<tag>).defeaturing("SliverFaces").detail();
model.geom(<tag>).defeaturing("SliverFaces").delete(<ftag>);
model.geom(<tag>).defeaturing("SliverFaces").deleteAll(<ftag>);
```

DESCRIPTION

`model.geom(`*`<tag>`*`).defeaturing("SliverFaces").delete(`*`<ftag>`*`)` creates a `DeleteSliverFaces` feature tagged *`<ftag>`* with the specified properties. The property `delete` is set to `selected`. If the feature can be built, it is inserted in the geometry sequence after the current feature; otherwise, the feature is discarded.

`model.geom(`*`<tag>`*`).defeaturing("SliverFaces").deleteAll(`*`<ftag>`*`)` works as the delete method, but the property `delete` is set to `all`.

It is also possible to create a `DeleteSliverFaces` feature using the standard `create` method. The following properties are available.

TABLE 4-11: AVAILABLE PROPERTIES

PROPERTY	VALUE	DEFAULT	DESCRIPTION
`delete`	`all` | `selected`	`selected`	Delete all sliver faces of given width, or a selection. Only available for the feature.
`entsize`	`double`	`1e-3`	Maximum face width.
`input`	`Selection`		Names of input objects.
`selresult`	`on` | `off`	`off`	Create selections of all resulting objects.
`selresultshow`	`all` | `obj` | `dom` | `bnd` | `edg` | `pnt` | `off`	`dom`	Show selections of resulting objects in physics, materials, and so on, or in part instances. `obj` is not available in a component's geometry. `dom`, `bnd`, and `edg` are not available in all features.
`contributeto`	String	`none`	Tag of cumulative selection to contribute to.

Sliver faces are narrow but long faces with large aspect ratio, which usually give rise to extremely fine local meshes in their vicinity.

`model.geom(`*`<tag>`*`).feature(`*`<ftag>`*`).find()` searches the input objects for faces with width less than `entsize`.

`model.geom(`*`<tag>`*`).feature(`*`<ftag>`*`).detail()` returns a selection object where you can select a subset of the faces found.

The `find` and `detail` methods of `model.geom(`*`<tag>`*`).defeaturing("SliverFaces")` has the corresponding functionality for the defeaturing tool.

Only faces that can be deleted without invalidating the object are deleted. If a face was not possible to delete, a warning message is given.

COMPATIBILITY

The following property is no longer supported:

TABLE 4-12: OBSOLETE PROPERTIES

PROPERTY	VALUE	DEFAULT	DESCRIPTION
Out	stx | ftx | ctx | ptx | status	none	Output variables

EXAMPLE

The following example imports the geometry model from the file defeaturing_demo_5.x_b, finds sliver faces narrower than $2 \cdot 10^{-3}$, and deletes the first of these.

```
Model model = ModelUtil.create("Model1");
model.geom().create("geom1",3);
model.geom("geom1").feature().create("imp1","Import");
model.geom("geom1").feature("imp1").set("filename",
     "defeaturing_demo_5.x_b");
model.geom("geom1").runAll();
model.geom("geom1").feature().create("dsl1","DeleteSliverFaces");
model.geom("geom1").feature("dsl1").selection("input").
set("imp1");
model.geom("geom1").feature("dsl1").set("entsize",2e-3);
model.geom("geom1").feature("dsl1").find();
model.geom("geom1").feature("dsl1").detail().setGroup(1);
model.geom("geom1").runAll();
```

SEE ALSO

DeleteFaces, DeleteSmallFaces

DeleteSmallFaces

Find and delete small faces in CAD objects.

SYNTAX

```
model.geom(gname).feature().create(<ftag>,"DeleteSmallFaces");
model.geom(<tag>).feature(<ftag>).selection(property);
model.geom(<tag>).feature(<ftag>).set(property,<value>);
model.geom(<tag>).feature(<ftag>).getType(property);
model.geom(<tag>).feature(<ftag>).find();
model.geom(<tag>).feature(<ftag>).detail();

model.geom(<tag>).defeaturing("SmallFaces").selection(property);
model.geom(<tag>).defeaturing("SmallFaces").set(property,<value>);
model.geom(<tag>).defeaturing("SmallFaces").find();
model.geom(<tag>).defeaturing("SmallFaces").detail();
model.geom(<tag>).defeaturing("SmallFaces").delete(<ftag>);
model.geom(<tag>).defeaturing("SmallFaces").deleteAll(<ftag>);
```

DESCRIPTION

`model.geom(<tag>).defeaturing("SmallFaces").delete(<ftag>)` creates a `DeleteSmallFaces` feature tagged *<ftag>* with the specified properties. The property `delete` is set to `selected`. If the feature can be built, it is inserted in the geometry sequence after the current feature; otherwise, the feature is discarded.

`model.geom(<tag>).defeaturing("SmallFaces").deleteAll(<ftag>)` works as the `delete` method, but the property `delete` is set to `all`.

It is also possible to create a `DeleteSmallFaces` feature using the standard `create` method. The following properties are available.

TABLE 4-13: AVAILABLE PROPERTIES

PROPERTY	VALUE	DEFAULT	DESCRIPTION
`delete`	`all` | `selected`	`selected`	Delete all small faces of given size, or a selection. Only available for the feature.
`entsize`	`double`	`1e-3`	Maximum face size.
`input`	`Selection`		Names of input objects.
`selresult`	`on` | `off`	`off`	Create selections of all resulting objects.

TABLE 4-13: AVAILABLE PROPERTIES

PROPERTY	VALUE	DEFAULT	DESCRIPTION
`selresultshow`	`all` | `obj` | `dom` | `bnd` | `edg` | `pnt` | `off`	`dom`	Show selections of resulting objects in physics, materials, and so on, or in part instances. `obj` is not available in a component's geometry. `dom`, `bnd`, and `edg` are not available in all features.
`contributeto`	String	`none`	Tag of cumulative selection to contribute to.

A small face is a face that fits within a sphere of specified radius, given in the property `entsize`.

`model.geom(<tag>).feature(<ftag>).find()` searches the input objects for faces with size less than `entsize`.

`model.geom(<tag>).feature(<ftag>).detail()` returns a selection object where you can select a subset of the faces found.

The `find` and `detail` methods of `model.geom(<tag>).defeaturing("SmallFaces")` has the corresponding functionality for the defeaturing tool.

Only faces that can be deleted without invalidating the object are deleted. If a face was not possible to delete, a warning message is given, accessible through `model.geom(<tag>).feature(<ftag>).problem()`.

COMPATIBILITY

The following property is no longer supported:

TABLE 4-14: OBSOLETE PROPERTIES

PROPERTY	VALUE	DEFAULT	DESCRIPTION
`Out`	`stx` | `ftx` | `ctx` | `ptx` | `status`	none	Output variables.

EXAMPLE

The following example imports the geometry model from the file `defeaturing_demo_6.x_b`, finds sliver faces narrower than 10^{-3}, and deletes the first of these.

```
Model model = ModelUtil.create("Model1");
model.geom().create("geom1",3);
model.geom("geom1").feature().create("imp1","Import");
```

```
model.geom("geom1").feature("imp1").set("filename",
     "defeaturing_demo_6.x_b");
model.geom("geom1").runAll();
model.geom("geom1").feature().create("df1","DeleteSmallFaces");
model.geom("geom1").feature("df1").selection("input").
      set("imp1");
model.geom("geom1").feature("df1").find();
model.geom("geom1").feature("df1").detail().setGroup(1);
model.geom("geom1").run();
```

SEE ALSO

DeleteFaces, DeleteSliverFaces

DeleteSpikes

Find and delete spikes in CAD objects.

SYNTAX

```
model.geom(<tag>).feature().create(<ftag>,"DeleteSpikes");
model.geom(<tag>).feature(<ftag>).selection(property);
model.geom(<tag>).feature(<ftag>).set(property,<value>);
model.geom(<tag>).feature(<ftag>).getType(property);
model.geom(<tag>).feature(<ftag>).find();
model.geom(<tag>).feature(<ftag>).detail();

model.geom(<tag>).defeaturing("Spikes").selection(property);
model.geom(<tag>).defeaturing("Spikes").set(property,<value>);
model.geom(<tag>).defeaturing("Spikes").find();
model.geom(<tag>).defeaturing("Spikes").detail();
model.geom(<tag>).defeaturing("Spikes").delete(<ftag>);
model.geom(<tag>).defeaturing("Spikes").deleteAll(<ftag>);
```

DESCRIPTION

`model.geom(<tag>).defeaturing("DeleteSpikes").delete(<ftag>)` creates a `DeleteSpikes` feature tagged *<ftag>* with the specified properties. The property `delete` is set to `selected`. If the feature can be built, it is inserted in the geometry sequence after the current feature; otherwise, the feature is discarded.

`model.geom(<tag>).defeaturing("DeleteSpikes").deleteAll(<ftag>)` works as the `delete` method, but the property `delete` is set to `all`.

It is also possible to create a `DeleteSpikes` feature using the standard `create` method. The following properties are available.

TABLE 4-15: AVAILABLE PROPERTIES

PROPERTY	VALUE	DEFAULT	DESCRIPTION
`delete`	`all` | `selected`	`selected`	Delete all spikes of given width, or a selection. Only available for the feature.
`entsize`	`double`	`1e-3`	Maximum spike width.
`input`	`Selection`		Names of input objects.
`selresult`	`on` | `off`	`off`	Create selections of all resulting objects.
`selresultshow`	`all` | `obj` | `dom` | `bnd` | `edg` | `pnt` | `off`	`dom`	Show selections of resulting objects in physics, materials, and so on, or in part instances. `obj` is not available in a component's geometry. `dom`, `bnd`, and `edg` are not available in all features.
`contributeto`	String	`none`	Tag of cumulative selection to contribute to.

A spike is a long and narrow protrusion on an edge or corner of a face defined by two or three edges.

`model.geom(`*<tag>*`).feature(`*<ftag>*`).find()` searches the input objects for spikes of width less than `entsize`.

`model.geom(`*<tag>*`).feature(`*<ftag>*`).detail()` returns a selection object where you can select a subset of the spikes found.

The find and detail methods of `model.geom(`*<tag>*`).defeaturing("Spikes")` has the corresponding functionality for the defeaturing tool.

Only spikes that can be deleted without invalidating the object are deleted. If a spike was not possible to delete, a warning message is given, accessible through `model.geom(`*<tag>*`).feature(`*<ftag>*`).problem()`.

COMPATIBILITY

The width of each spike is no longer returned.

The following property is no longer supported:

TABLE 4-16: OBSOLETE PROPERTIES

PROPERTY	VALUE	DEFAULT	DESCRIPTION
Out	stx | ftx | ctx | ptx | status	none	Output variables.

EXAMPLE

The following example imports the geometry model from the file defeaturing_demo_7.x_b, finds all spikes narrower than 10^{-4}, and deletes the first of these.

```
Model model = ModelUtil.create("Model1");
model.geom().create("geom1",3);
model.geom("geom1").feature().create("imp1","Import");
model.geom("geom1").feature("imp1").set("filename",
     "defeaturing_demo_7.x_b");
model.geom("geom1").runAll();
model.geom("geom1").feature().create("dsp1","DeleteSpikes");
model.geom("geom1").feature("dsp1").selection("input").
      set("imp1");
model.geom("geom1").feature("dsp1").set("entsize",1e-4);
model.geom("geom1").feature("dsp1").find();
model.geom("geom1").feature("dsp1").detail().setGroup(1);
model.geom("geom1").runAll();
```

SEE ALSO

DeleteShortEdges, DeleteSliverFaces

DetachFaces

Detach faces from CAD objects to form a new (child) solid.

SYNTAX

```
model.geom(<tag>).feature().create(<ftag>,"DetachFaces");
model.geom(<tag>).feature(<ftag>).selection(property);
model.geom(<tag>).feature(<ftag>).set(property,<value>);
model.geom(<tag>).feature(<ftag>).getType(property);

model.geom(<tag>).defeaturing("DetachFaces").selection(property);
model.geom(<tag>).defeaturing("DetachFaces").set(property,<value>);
model.geom(<tag>).defeaturing("DetachFaces").delete(<ftag>);
```

DESCRIPTION

`model.geom(<tag>).defeaturing("DetachFaces").delete(<ftag>)` creates a `DetachFaces` feature tagged *<ftag>* with the specified properties. If the feature can be built, it is inserted in the geometry sequence after the current feature; otherwise, the feature is discarded.

It is also possible to create a `DetachFaces` feature using the standard `create` method.

TABLE 4-17: AVAILABLE PROPERTIES

PROPERTY	VALUE	DEFAULT	DESCRIPTION
`input`	`Selection`		Faces to detach.
`healchild`	`fill` | `patchchild` | `patchparent`	`patchparent`	Healing method used on the child object.
`healparent`	`fill` | `patch`	`patch`	Healing method used on the parent object.
`selresult`	`on` | `off`	`off`	Create selections of all resulting objects.
`selresultshow`	`all` | `obj` | `dom` | `bnd` | `edg` | `pnt` | `off`	`dom`	Show selections of resulting objects in physics, materials, and so on, or in part instances. `obj` is not available in a component's geometry. `dom`, `bnd`, and `edg` are not available in all features.
`contributeto`	String	`none`	Tag of cumulative selection to contribute to.

The faces in the property `input` are detached from their *parent* object. A new solid, the *child* object, are formed from the detached faces. The output objects are the healed parent and child objects.

The property `healparent` determines how the parent object is healed to form a new solid after detaching the faces. The value `fill` means that a new face is formed based on the surrounding edges of each wound. The value `patch` means that the surrounding faces of each wound are grown or shrunk.

The property `healchild` determines how the child solid is constructed from the detached faces. The value `fill` means that a new face is formed based on the surrounding edges of each wound. The value `patchchild` means that the detached faces are grown or shrunk to form a solid. The value `patchparent` means that the parent faces surrounding the detached faces are grown or shrunk to form a solid together with the detached faces.

EXAMPLE

The following example imports the COMSOL Multiphysics geometry file `defeaturing_demo_2.mphbin` and detaches a hole defined by a set of faces:

```
Model model = ModelUtil.create("Model1");
model.geom().create("geom1",3);
model.geom("geom1").feature().create("imp1","Import");
model.geom("geom1").feature("imp1").set("filename",
     "defeaturing_demo_2.mphbin");
model.geom("geom1").runAll();
model.geom("geom1").feature().create("det1","DetachFaces");
model.geom("geom1").feature("det1").selection("input").
      set("imp1",6,7,8,9,11,12,13);
model.geom("geom1").runAll();
```

COMPATIBILITY

The following property is no longer supported:

TABLE 4-18: OBSOLETE PROPERTIES

PROPERTY	VALUE	DEFAULT	DESCRIPTION
`Out`	`stx` | `ftx` | `ctx` | `ptx`	none	Output variables

SEE ALSO

`DeleteFaces`

Export, ExportFinal

Using the CAD Import Module, Design Module, or a LiveLink product for CAD software, export selected geometry objects or the finalized geometry to a 3D CAD format, such as ACIS, Parasolid, STEP, and IGES.

To export selected geometry objects to a file, first select the objects to export using

```
model.component(<ctag>).geom(<tag>).export().selection().set(<obj
names>);
```

where *<objnames>* is a string array of object names. Then export them by entering

```
model.component(<ctag>).geom(<tag>).export(<filename>);
```

To export the finalized geometry to a file, enter

```
model.component(<ctag>).geom(<tag>).exportFinal(<filename>);
```

where *<filename>* is a string.

In the above commands the file extension in the *<filename>* string determines the file format, which can be of any of the following:

TABLE 4-19: SUPPORTED FILE FORMATS

FILE FORMAT	FILE EXTENSION
Parasolid Binary (3D)	`.x_b`, `.xmt_bin`
Parasolid Text (3D)	`.x_t`, `.xmt_txt`
ACIS Binary (3D)	`.sab`
ACIS Text (3D)	`.sat`
IGES File (3D)	`.igs`, `.iges`
STEP File (3D)	`.step`, `.stp`

EXPORTING TO AN ACIS FILE

When exporting to an ACIS file you can set the ACIS file format version using

```
model.component(<ctag>).geom(<tag>).export().setAcisVersion(<vers
ion>);
```

where *<version>* is a string `4.0`, `7.0`, or `2016 1.0`. Default is `2016 1.0`.

EXPORTING TO A PARASOLID FILE

The Parasolid text or binary file generated by the export is of version 31.0.

When exporting to a Parasolid format, a unit conversion can optionally be performed during export. Use the following method to select the export length unit:

```
model.component(<ctag>).geom(<tag>).export().setLengthUnit(<unit>
);
```

where *<unit>* is either `fromgeom` (default) to disable unit conversion or a COMSOL Multiphysics length unit, such as `m` for meters or `in` for inches. To get the current value of the export length unit type:

```
model.component(<ctag>).geom(<tag>).export().getLengthUnit();
```

To decide how the non-manifold objects are exported use the following method:

```
model.component(<ctag>).geom(<tag>).export().setSplitInManifold(<
value>);
```

where *<value>* is either `true` (default) to split the objects into manifold objects during the export, or `false` to export the unmodified objects.

EXPORTING TO AN IGES FILE

When exporting to the IGES format, a unit conversion can optionally be performed during export. Use the following method to select the export length unit:

```
model.component(<ctag>).geom(<tag>).export().setLengthUnitIGES(<unit>);
```

where *<unit>* is either `fromgeom` (default) to disable unit conversion or a supported length unit: `uin`, `um`, `mil`, `mm`, `cm`, `in`, `ft`, `m`, `km`, `mi`.To get the current value of the export length unit type:

```
model.component(<ctag>).geom(<tag>).export().getLengthUnitIGES();
```

EXPORTING TO A STEP FILE

When exporting to the STEP format, a unit conversion can optionally be performed during export. Use the following method to select the export length unit:

```
model.component(<ctag>).geom(<tag>).export().setLengthUnitSTEP(<unit>);
```

where *<unit>* is either `fromgeom` (default) to disable unit conversion or a supported length unit: `nm`, `uin`, `um`, `mil`, `mm`, `cm`, `in`, `dm`, `ft`, `m`, `km`, `mi`. To get the current value of the export length unit type:

```
model.component(<ctag>).geom(<tag>).export().getLengthUnitSTEP();
```

SEE ALSO

`Import`

Import

Import geometry objects from a 3D CAD file using the CAD Import Module, Design Module, or a LiveLink product for CAD software.

SYNTAX

```
model.geom(<tag>).feature().create(<ftag>,"Import");
model.geom(<tag>).feature(<ftag>).set(property,<value>);
model.geom(<tag>).feature(<ftag>).getType(property);
model.geom(<tag>).feature(<ftag>).importData();
```

DESCRIPTION

`model.geom(<tag>).feature().create(<ftag>,"Import")` creates an import feature. When the property `filename` is set to a filename recognized as a 3D CAD file, the property `type` is set to `cad`. The following properties are available.

TABLE 4-20: AVAILABLE PROPERTIES

PROPERTY	VALUE	DEFAULT	DESCRIPTION
`check`	`on` | `off`		Check imported objects for errors.
`filename`	`String`		Filename.
`fillholes`	`on` | `off`	`off`	Attempt to generate new faces to replace missing geometry if the property knit is solid or surface
`importtol`	`double`	`1e-5`	Absolute repair tolerance.
`keepbnd`	`on` | `off`	`on`	Import surface objects.
`keepfree`	`on` | `off`	`off`	Import curve and point objects.
`keepsolid`	`on` | `off`	`on`	Import solid objects.
`knit`	`solid` | `surface` | `off`	`solid`	Knit together surface objects to form solids or surface objects.
`removeredundant`	`on` | `off`	`off`	Remove redundant edges and vertices.
`repair`	`on` | `off`	`on`	Repair imported objects.
`type`	`cad`		Type of import.
`unit`	`source` | `current`	`source`	Take length unit from file or from the current geometry unit.
`unitecurves`	`on` | `off`	`on`	Unite curve objects.
`selresult`	`on` | `off`	`off`	Create selections of all resulting objects.

TABLE 4-20: AVAILABLE PROPERTIES

PROPERTY	VALUE	DEFAULT	DESCRIPTION
`selresultshow`	`all` | `obj` | `dom` | `bnd` | `edg` | `pnt` | `off`	`dom`	Show selections of resulting objects in physics, materials, and so on, or in part instances. `obj` is not available in a component's geometry. `dom`, `bnd`, and `edg` are not available in all features.
`contributeto`	String	`none`	Tag of cumulative selection to contribute to.

The file to import is specified by `filename`, which can have of any of the following formats:

TABLE 4-21: SUPPORTED 3D CAD FILE FORMATS

FILE FORMAT	NOTE	FILE EXTENSION
Autodesk Inventor	1, 3	`.ipt, .iam`
CATIA V5	2, 3	`.CATPart, .CATProduct`
IGES	1	`.igs, .iges`
Parasolid	1	`.x_t, .x_b`
PTC Pro/ENGINEER	1	`.prt, .asm`
SAT (ACIS)	1	`.sat, .sab`
SOLIDWORKS	1, 3	`.sldprt, .sldasm`
STEP	1	`.step, .stp`

Note 1: This format requires a license for the CAD Import Module, Design Module, or a LiveLink product for a CAD package.

Note 2: This format requires, in addition to the CAD Import Module, Design Module, or a LiveLink product for a CAD package, a license for the File Import for CATIA V5 module.

Note 3: Only supported on Windows.

The imported geometry objects are represented using the Parasolid geometry kernel, which is the geometry kernel utilized by the CAD Import Module and the LiveLink products for CAD software.

The method

```
model.geom(gname).feature(<ftag>).importData()
```

imports the file again, even if the feature is built.

The import can generate object, boundary, edge, and point selections based on material, layer, and color assignments in the 3D CAD file. The following properties are available for working these selections:

TABLE 4-22: AVAILABLE PROPERTIES

PROPERTY	VALUE	DEFAULT	DESCRIPTION
`selcadshownamesfromfileobj`	boolean	`false`	Show the object selection names from the file in the GUI.
`selcadnameobj`	String[]	empty	Names of object selections in 3D CAD import.
`selcadnameinfileobj`	String[]	empty	Original names of object selections in 3D CAD import. Read-only.
`selcadkeepobj`	`on | off`	empty	Keep object selections in 3D CAD import.
`selcadshowobj`	`on | off`	empty	Show object selections in 3D CAD import in physics, materials, and so on; in part instances; or in 3D from a plane geometry.
`selcadcontributetoobj`	String[]	empty	Tags of cumulative selection to contribute to (or none to not contribute), for object selections in 3D CAD import.
`selcadtagobj`	String[]	empty	Tags of object selections (read-only, hidden in GUI) in 3D CAD import.

TABLE 4-22: AVAILABLE PROPERTIES

PROPERTY	VALUE	DEFAULT	DESCRIPTION
`selcadcolorobj`	String[]	empty	Colors of object selections (read-only) in 3D CAD import. The color is stored as a comma-separated triple of numbers between 0 and 1. It can also be "none" (in which case it will be displayed in yellow).
`selindividualintable`	boolean	`false`	Show individual object selections and, for the knit case, individual original object selections in the CAD-tables.
`selcadshownamesfromfilebnd`	boolean	`false`	Show the boundary selection names from the file in the GUI.
`selcadnamebnd`	String[]	empty	Names of boundary selections in 3D CAD import.
`selcadnameinfilebnd`	String[]	empty	Original names of boundary selections in 3D CAD import. Read-only.
`selcadkeepbnd`	`on | off`	empty	Keep boundary selections in 3D CAD import.
`selcadshowbnd`	`on | off`	empty	Show boundary selections in 3D CAD import in physics, materials, and so on; in part instances; or in 3D from a plane geometry.
`selcadcontributetobnd`	String[]	empty	Tags of cumulative selection to contribute to (or none to not contribute), for boundary selections in 3D CAD import.

TABLE 4-22: AVAILABLE PROPERTIES

PROPERTY	VALUE	DEFAULT	DESCRIPTION
`selcadtagbnd`	String[]	empty	Tags of boundary selections (read-only, hidden in GUI) in 3D CAD import.
`selcadcolorbnd`	String[]	empty	Colors of boundary selections (read-only) in 3D CAD import. The color is stored as a comma-separated triple of numbers between 0 and 1. It can also be "none" (in which case it will be displayed in yellow).
`selcadshownamesfromfileedg`	boolean	`false`	Show the edge selection names from the file in the GUI.
`selcadnameedg`	String[]	empty	Names of edge selections in 3D CAD import.
`selcadnameinfileedg`	String[]	empty	Original names of edge selections in 3D CAD import. Read-only.
`selcadkeepedg`	`on` | `off`	empty	Keep edge selections in 3D CAD import.
`selcadshowedg`	`on` | `off`	empty	Show edge selections in 3D CAD import in physics, materials, and so on; in part instances; or in 3D from a plane geometry.
`selcadcontributetoedg`	String[]	empty	Tags of cumulative selection to contribute to (or none to not contribute), for edge selections in 3D CAD import.
`selcadtagedg`	String[]	empty	Tags of edge selections (read-only, hidden in GUI) in 3D CAD import.

TABLE 4-22: AVAILABLE PROPERTIES

PROPERTY	VALUE	DEFAULT	DESCRIPTION
`selcadcoloredg`	String[]	empty	Colors of edge selections (read-only) in 3D CAD import. The color is stored as a comma-separated triple of numbers between 0 and 1. It can also be "none" (in which case it will be displayed in yellow).
`selcadshownamesfromfilepnt`	boolean	`false`	Show the point selection names from the file in the GUI.
`selcadnamepnt`	String[]	empty	Names of point selections in 3D CAD import.
`selcadnameinfilepnt`	String[]	empty	Original names of point selections in 3D CAD import. Read-only.
`selcadkeeppnt`	`on | off`	empty	Keep point selections in 3D CAD import.
`selcadshowpnt`	`on | off`	empty	Show point selections in 3D CAD import in physics, materials, and so on; in part instances; or in 3D from a plane geometry.
`selcadcontributetopnt`	String[]	empty	Tags of cumulative selection to contribute to (or none to not contribute), for point selections in 3D CAD import.

TABLE 4-22: AVAILABLE PROPERTIES

PROPERTY	VALUE	DEFAULT	DESCRIPTION
`selcadtagpnt`	String[]	empty	Tags of point selections (read-only, hidden in GUI) in 3D CAD import.
`selcadcolorpnt`	String[]	empty	Colors of point selections (read-only) in 3D CAD import. The color is stored as a comma-separated triple of numbers between 0 and 1. It can also be "none" (in which case it will be displayed in yellow).

COMPATIBILITY

The following property is no longer supported:

TABLE 4-23: OBSOLETE PROPERTIES

PROPERTY	VALUE	DEFAULT	DESCRIPTION
`coercion`	`solid` | `face` | `off`	solid	Alias for knit. face is equivalent to surface.

SEE ALSO

`Export`, `ExportFinal`

Knit

Knit surface CAD objects to form solids or surface objects.

SYNTAX

```
model.geom(<tag>).feature().create(<ftag>,"Knit");
model.geom(<tag>).feature(<ftag>).selection(property);
model.geom(<tag>).feature(<ftag>).set(property,<value>);
model.geom(<tag>).feature(<ftag>).getType(property)
```

DESCRIPTION

`model.geom(<ftag>).feature().create(<ftag>,"Knit")` creates a knit feature tagged *<ftag>*. The following properties are available.

TABLE 4-24: AVAILABLE PROPERTIES

PROPERTY	VALUE	DEFAULT	DESCRIPTION
`fillholes`	`on | off`	`off`	Attempt to generate new faces to replace missing geometry
`input`	`Selection`		Names of input surface objects.
`repairtol`	`double`	`1e-5`	Absolute repair tolerance.
`selresult`	`on | off`	`off`	Create selections of all resulting objects.
`selresultshow`	`all | obj | dom | bnd | edg | pnt | off`	`dom`	Show selections of resulting objects in physics, materials, and so on, or in part instances. `obj` is not available in a component's geometry. `dom`, `bnd`, and `edg` are not available in all features.
`contributeto`	String	`none`	Tag of cumulative selection to contribute to.

This function also removes gaps and spikes that are within the absolute tolerance specified in the property `repairtol`.

COMPATIBILITY

The following property is no longer supported:

TABLE 4-25: OBSOLETE PROPERTIES

PROPERTY	VALUE	DEFAULT	DESCRIPTION
`Out`	`stx | ftx | ctx | ptx`	none	Output variables.

EXAMPLE

The following example imports the file `repair_demo_2.x_b`, and knits the surface objects into a solid. A gap is also removed during the operation.

```
Model model = ModelUtil.create("Model1");
model.geom().create("geom1",3);
model.geom("geom1").feature().create("imp1","Import");
model.geom("geom1").feature("imp1").set("filename",
                                        "repair_demo_2.x_b");
model.geom("geom1").runAll();
model.geom("geom1").feature().create("knit1","Knit");
```

```
model.geom("geom1").feature("knit1").selection("input").
                                         set("imp1");
model.geom("geom1").feature("knit1").set("repairtol",1e-3);
model.geom("geom1").runAll();
```

SEE ALSO

Repair

LiveLinkRevit

Synchronize geometry objects with a Revit project.

SYNTAX

```
model.geom(<tag>).feature().create(<ftag>,"LiveLinkRevit");
model.geom(<tag>).feature(<ftag>).set(property,<value>);
model.geom(<tag>).feature(<ftag>).getType(property);
model.geom(<tag>).feature(<ftag>).updateCadParamTable(add,repl);
model.geom(<tag>).feature(<ftag>).importData();
```

DESCRIPTION

`model.geom(<tag>).feature().create(<ftag>,"LiveLinkRevit")` creates a `LiveLinkRevit` feature. The following properties are available.

TABLE 4-26: AVAILABLE PROPERTIES

PROPERTY	VALUE	DEFAULT	DESCRIPTION
`check`	`on	off`		Check imported objects for errors.
`configuration`	String		The synchronized configuration of the Revit project.
`document`	String		The full path of the synchronized Revit project.
`importtol`	`double`	`1e-5`	Absolute repair tolerance.
`keepbnd`	`on	off`	`on`	Import surface objects.
`keepsolid`	`on	off`	`on`	Import solid objects.
`param`	String[]		Name of parameters to set in Revit. Only parameters with sync set to on are sent.
`paramexpr`	String[]		Values of parameters to send to Revit.

TABLE 4-26: AVAILABLE PROPERTIES

PROPERTY	VALUE	DEFAULT	DESCRIPTION
removeredundant	on | off	on	Remove redundant edges and vertices.
repair	on | off	off	Repair imported objects.
selname	String[]		Read only property that corresponds to the names of the resulting selections.
seltag	String[]		Read only property that corresponds to the tags of the resulting selections.
sync	String[]		Enable/disable synchronization of parameters. Valid values are on or off.
synchronizewith	active | specified	active	Synchronize the active project in Revit or the specified project.
unit	source | current	source	Take length unit from Revit, or from the current geometry unit.

The method

```
model.geom(<tag>).feature(<ftag>).importData();
```

generates the geometry objects for the project in Revit Architecture according to the synchronization settings, and sends back geometry objects to COMSOL.

The method

```
model.geom(<tag>).feature(<ftag>).updateCadParamTable(add,repl);
```

updates the properties `param` and `paramexpr` with data read from Revit. If the add argument is true, all parameter names retrieved from Revit that do not already exist in param are appended to param, corresponding COMSOL variable names are appended to paramexpr and the corresponding values are added to the COMSOL global parameters table. If the repl argument is true, all parameters already present in param that also have sync set to on have their corresponding values replaced by the values retrieved from Revit. If paramexpr has a value equal to a COMSOL parameter, the value of that parameter is replaced. Otherwise, if paramexpr is a numerical value, possibly with unit, the value of paramexpr is replaced.

The imported geometry objects are represented using the Parasolid geometry kernel, which is the geometry kernel utilized by the CAD Import Module, Design Module, and the LiveLink products for CAD software.

Repair

Repair CAD objects.

SYNTAX

```
model.geom(<tag>).feature().create(<ftag>,"Repair");
model.geom(<tag>).feature(<ftag>).selection(property);
model.geom(<tag>).feature(<ftag>).set(property,<value>);
model.geom(<tag>).feature(<ftag>).getType(property)
```

DESCRIPTION

`model.geom(<tag>).feature().create(<ftag>,"Repair")` creates a repair feature tagged *<ftag>*. The following properties are available.

TABLE 4-27: AVAILABLE PROPERTIES

PROPERTY	VALUE	DEFAULT	DESCRIPTION
`input`	`Selection`		Names of input surface objects
`repairtol`	`double`	`1e-5`	Absolute repair tolerance
`selresult`	`on` | `off`	`off`	Create selections of all resulting objects.
`selresultshow`	`all` | `obj` | `dom` | `bnd` | `edg` | `pnt` | `off`	`dom`	Show selections of resulting objects in physics, materials, and so on, or in part instances. `obj` is not available in a component's geometry. `dom`, `bnd`, and `edg` are not available in all features.
`simplify`	`on` | `off`	`off`	Simplify the underlying curve and surface manifolds of geometric entities
`contributeto`	String	`none`	Tag of cumulative selection to contribute to.

The function tries to remove or repair the following defects:

- Entities with invalid sense
- Invalid edge and vertex tolerances
- Invalid manifolds

- Self-intersecting manifolds
- Non-G1 manifolds
- Missing edge or vertex manifolds
- Missing vertex
- Vertices not on curve of edge
- Edges and vertices not on surface of face
- Removal of surface self-intersections that lie outside the face
- Splitting at edge intersections which have no vertex
- Removal of discontinuities by either splitting or smoothing
- Remove small features (short edges, small faces, sliver faces, and spikes)

COMPATIBILITY

The following property is no longer supported:

TABLE 4-28: OBSOLETE PROPERTIES

PROPERTY	VALUE	DEFAULT	DESCRIPTION
`Out`	`stx` | `ftx` | `ctx` | `ptx`	none	Output variables

EXAMPLE

The following example imports the file `repair_demo_2.x_b`, and repairs the resulting objects.

```
Model model = ModelUtil.create("Model1");
model.geom().create("geom1",3);
model.geom("geom1").feature().create("imp1","Import");
model.geom("geom1").feature("imp1").set("filename",
                                        "repair_demo_2.x_b");
model.geom("geom1").runAll();
model.geom("geom1").feature().create("rep1","Repair");
model.geom("geom1").feature("rep1").selection("input").
                                    set("imp1");
model.geom("geom1").feature("rep1").set("repairtol",1e-3);
model.geom("geom1").runAll();
```

SEE ALSO

`Knit`